N68-23987

SOLAR FLAT PLATE

FINAL REPORT

HIT-206

October 1966

HITTMAN ASSOCIATES, INC.
BALTIMORE, MARYLAND

LEGAL NOTICE

FOREWORD

The analysis and computer program described herein was prepared under Contract Number NAS5-9167 for the National Aeronautics and Space Administration, Goddard Space Flight Center.

TABLE OF CONTENTS

SUMMARY

Solar thermoelectric energy conversion panels may be constructed from modules, each of which consists of an absorber, a thermoelement or thermoelectric couple, and a radiator. The absorber collects solar heat which causes a high temperature to be established. The thermoelements convert a portion of this heat to electricity, and the unconverted heat is rejected to space by the radiator.

Analyses conducted prior to the work reported herein have been limited to mathematical models which could be handled using manual calculation techniques. Calculation accuracy has consequently not been as precise as desired. Therefore, an accurate analysis has been developed, reported herein, so that the performance characteristics of the solar flat plate may be predicted for expected operational conditions. The mathematical model obtained as a result of this study contains a number of non-linear second order differential equations which cannot be solved analytically. A digital computer program therefore has been written for the IBM-7094 so that a solution may be obtained.

Major emphasis has been placed in the writing of the program so that calculation accuracy and ease of use of the program will result. The calculation accuracy has been obtained through a careful selection of the assumptions necessary to define the mathematical model and by careful computation techniques. Ease of use has been obtained through the use of the simplified input routines and carefully presented data. The input routine which has been developed for this program is virtually independent of format and the required data may be read in almost any order. Parametric studies may be performed by reading data which have been changed from previous runs while data which have not been changed need not be reread. Automatic generation of all of the mesh information is taken care of by the computer program with an input definition of the number of points around the periphery of the solar flat plate module. Computer output options exist so that limited or detailed computed results may be obtained. If desired, the output may contain a map of the computer node points printed to a scale of one part in 100, as well as temperatures and electrical characteristics as functions of time.

A comparison of the computer program results with experimental and manual calculations has been obtained. These show good correlation with the limited experimental information which is available, and predict slightly lower performance characteristics than have been obtained using manual approaches. In general, the correlation is good.

A limited parametric study has been performed and reported herein which shows characteristics to be expected for earth orbit behavior and for an application close to the sun.

I. INTRODUCTION

A. Background

The satisfaction of auxiliary power requirements in space constitutes a difficult problem. To date, most of these requirements have been met by the direct conversion of solar radiation to electricity. Until recently, the only practical solid-state device available for this purpose was the photovoltaic cell. Recent developmental work has resulted in the availability of an alternate technique, the thermoelectric flat plate solar generator. This electrical supply has promise of more attractive cost per watt of electrical power produced, lower weight/watt of power produced, and has high resistance to nuclear radiation. The solar flat plate also appears attractive for close-in orbits to the sun where at present photovoltaic cells would be difficult to use.

The background technology and a number of preliminary calculations illustrating characteristics are discussed by Fuschillo,* and will not be considered in detail in this report. The purpose of this report is to present an investigation of the theoretical performance of the device, to compare the results of the theoretical investigation with experimentally determined performance, and to present a parametric study which may be utilized to evaluate use of the device for various postulated missions.

B. The Solar Flat Plate Thermoelectric Generator

The solar thermoelectric generator consists of three major items: an absorber, thermoelements and a radiator. The absorber consists of a thin plate coated to give a high absorptivity for solar radiation with a low emissivity. This is roughly sun-oriented and attains a relatively high temperature because of its absorption-emissive properties. This heats the hot end of the thermoelements which are attached to the back of the absorber. The other end of the thermoelement is attached to a radiator, which has an exposed surface with a high emissivity coating so that unconverted heat can be rejected to space. The internal surfaces which do not "see" space are normally highly polished to minimize radiant interchange between the absorber and radiator.

Normal practice is to construct the solar panel so that the radiator and absorber serve as electrical interconnections between the thermoelements. Any desired arrays of series or series-parallel arrangements may be immediately obtained. A photograph of a typical panel is shown in Figure I-1. Construction details may be seen in Figure I-2 which shows the components of a typical cell.

---

* Fuschillo, N., Gibson, R., Eggelston, F. K., and Epstein, J., "Flat Plate Solar Electric Generator for Nuclear Earth Orbits", Advanced Energy Conversion, Vol. 6, pp. 103-125, 1966.

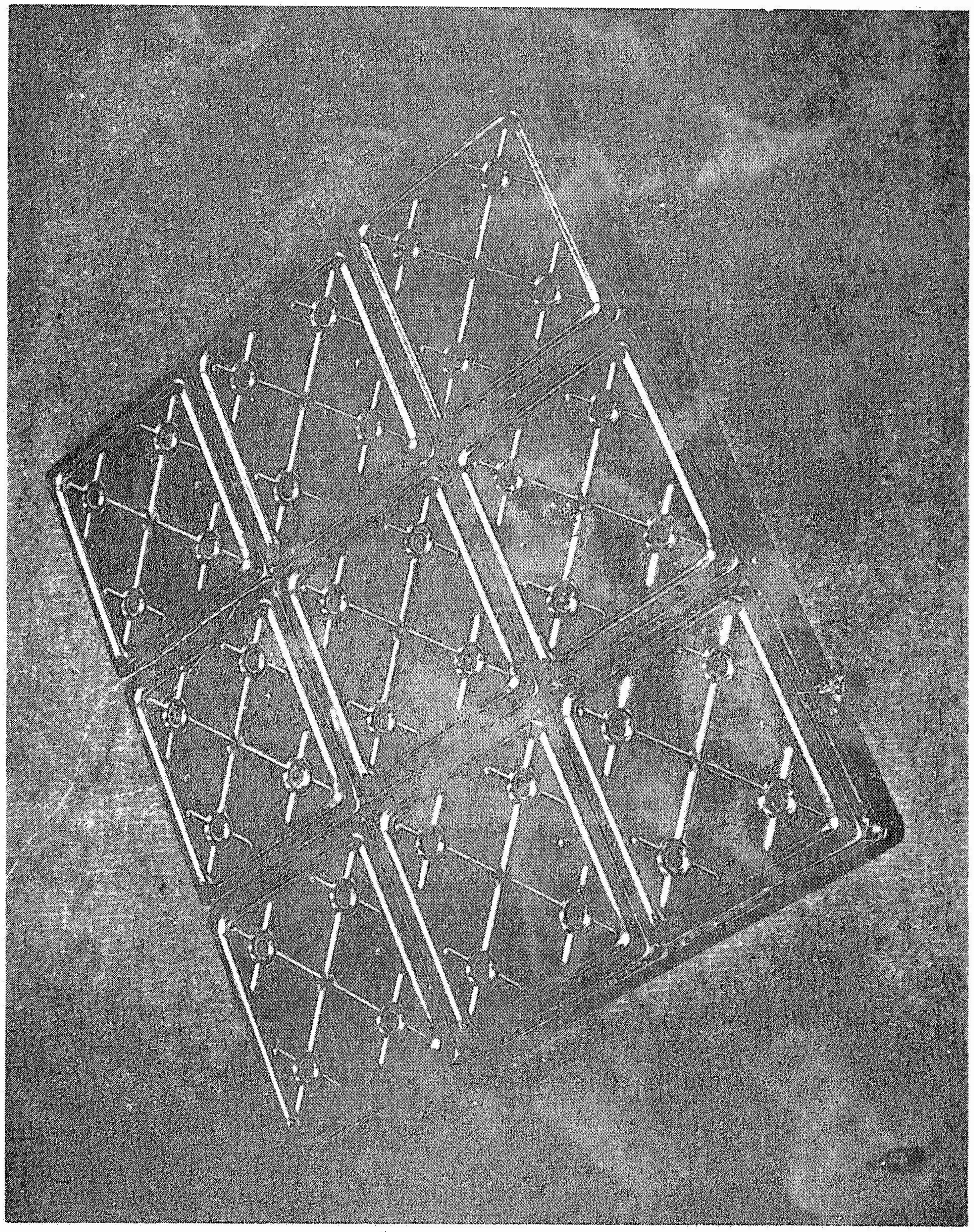


Figure I-1. Flat Plate Solar Thermoelectric Generator Panel

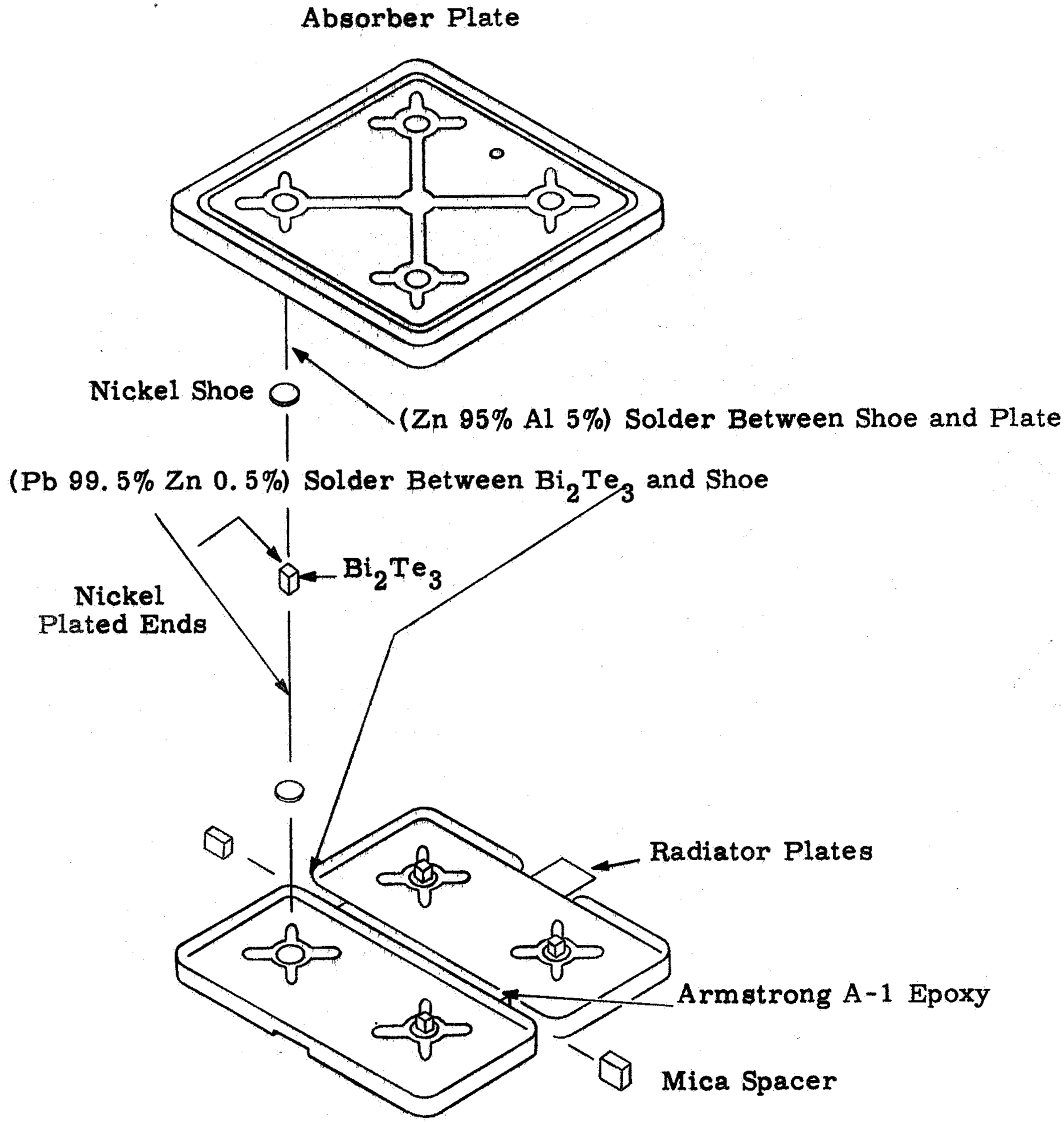


Figure I-2. Unit Couple Assembly Drawing

II. PANEL ANALYSIS

Two basic types of analyses are necessary to fully investigate the solar flat plate capabilities. Long term behavior of the type associated with a constant solar flux and constant heat rejection characteristics are of interest so that the steady state performance may be investigated. Power generation behavior associated with transient type phenomena are required for such applications as a non-oriented array mounted on a spinning satellite or to predict the characteristics when a satellite moves from shade to sun conditions or vice versa. Either of these basic investigations may be performed by considering the three components comprising the system: the thermoelements, the absorber and the radiator.

Thermoelectric theory is relatively well known from the analysis or engineering standpoint and will accurately predict behavior provided suitable mathematical models are selected. The absorber and radiator may both be described utilizing transient heat conduction theory with appropriate incorporation of the radiant heat transfer phenomena. The basic assumptions required are as follows:

(1) The radiator and absorber may be treated as transient two-dimensional heat conductors with a variable heat source and no temperature gradient perpendicular to the surface.

(2) Heat input to the absorber is due to the solar flux, planet albedo, and an effective planet temperature with full recognition given to the various absorptivity and view factor terms.

(3) Radiator input is treated in a manner similar to the absorber.

(4) Thermal coupling between the absorber and radiator occurs due to the thermoelements and radiant heat transfer between the two plates.

(5) The thermoelement can be described by a transient one-dimensional differential equation which contains terms for heat conduction, Joule heating, the Peltier, Seebeck and Thomson effects, and radiant interchange between the thermoelement, absorber and radiator.

(6) The effect of resistance between a thermoelement and mounting hardware is included as a "contact" resistance (which must be determined experimentally).

(7) Heat generation within the plate due to the flow of electricity and the effect of heat received by the plate from the thermoelements is small and may be neglected.

The justification for these assumptions and the mathematical model which results are discussed in the following sections.

A. Thermoelectric Analysis

Analysis of thermoelectric elements is generally achieved by taking a heat balance over the entire thermoelectric couple to determine the heat required at the hot junction. If this is then divided into the output power, the efficiency is obtained. Suitable differentiations are then commonly used to obtain an optimum efficiency. It is relatively common practice to ignore the contact resistance in this approach, or to make assumptions pertaining to the behavior of contact resistance with cross-sectional area. If for the moment the correctness or incorrectness of these assumptions is neglected, one problem involved with this approach is the difficulty of understanding the errors involved in obtaining the heat balance. A further problem is that this approach may be used only for the steady state, and is inappropriate for use with transient analysis. One of the first analyses which considered the steady state thermoelectric behavior from a microscopic standpoint was presented by Lyon and Bustard.* Although this analysis was derived on the basis of non-transient conditions, it is a relatively simple extension to utilize it for the transient representation. The approach will be used herein because it has been shown to be accurate.

Although the assumption of Lyon and Bustard that the variation of contact resistance with the inverse of cross sectional area is good for some analyses, it is not a legitimate assumption for all cases. The error is expected to become larger as thermoelectric element size is decreased, and since size will be of importance in some of the applications of the solar flat plate, the assumption is not considered legitimate herein.

To obtain a mathematical model which describes thermoelectric behavior, we first consider the thermoelectric phenomena which are involved. These are:

(1) Thermal conduction: the transport of heat from one position in the thermoelectric material to another due to the temperature gradient.

(2) Joule heating: the internal heating effect within the thermoelectric material due to resistance to the flow of electricity.

(3) Seebeck effect: the phenomenon of the flow of electricity that occurs in a closed loop which is composed of two dissimilar materials when the junction between the materials is maintained at different temperatures.

(4) Thomson effect: the phenomenon of heat generation or absorption which takes place when an electrical current flows in the presence of a temperature gradient.

---

* Lyon, W. C, and Bustard, T. S., "Digital Computation of Thermoelement Operational Parameters", Advanced Energy Conversion, Vol. 2, 1962, pp. 197-208.

(5) Peltier effect: the heat generation or absorption phenomenon that occurs at a junction between dissimilar materials when an electrical current is caused to flow.

Incorporation of these well understood theoretical effects is sufficient to describe the thermoelectric behavior if no other perturbations occur. However, any mathematical analysis must include the effects of the surroundings to a greater or a lesser extent and must also consider the actual physical situation in which the thermoelements are manufactured or used. Consequently, further assumptions are necessary. These are:

(1) The thermoelectric material is homogeneous.

(2) Heat interchange between the lateral or exposed surface of the thermoelements may be represented by considering only radiant heat transfer between the thermoelement surface and the surroundings as seen by the surface.

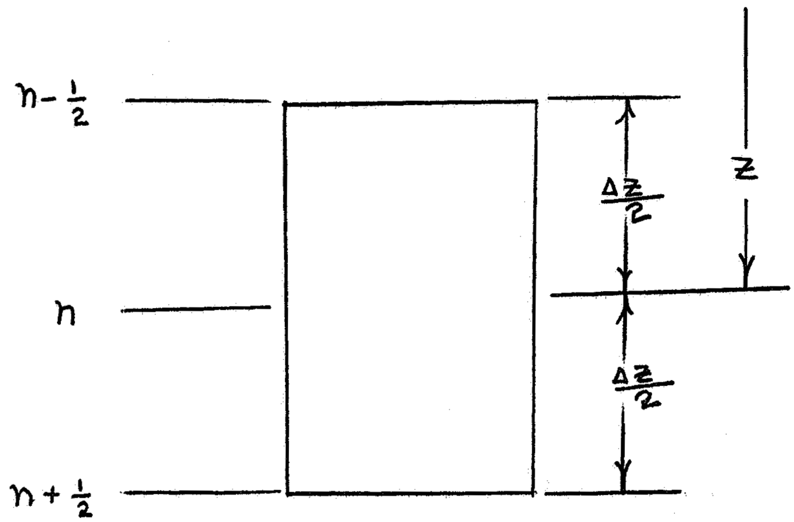
(3) There is no temperature gradient within the thermoelement normal to the axis.

Homogeneity of material is generally obtained to the degree necessary for a valid mathematical analysis. This assumption is basically necessary at this time regardless of the degree of homogeneity or non-homogeneity in order that a mathematical representation can be obtained. However, a good thermoelectric material is homogeneous, and it is expected that little error will be involved with this assumption.

The assumption that the thermoelement exchanges heat from the exposed surface with its surroundings is in conflict with the assumption that no temperature gradient exists within the thermoelement normal to the axis. Obviously, if heat is going to be rejected from the lateral surface, there must be a gradient with respect to a normal drawn to that surface. This assumption is justified, however, because the quantity of heat which is conducted along the thermoelement is very much greater than that which is radiantly interchanged between the thermoelement and the absorber-radiator. The error involved is consequently small.

The calculation of the energy interchange between the thermoelement surface and its surroundings could become extremely complicated, with a consequent result that a tremendous amount of time would be used in obtaining numerical results. However, the thermoelement basically "sees" only the absorber and the radiator inside surfaces. If these are assumed to be represented by average temperatures, then an approximation of the heat transfer may be readily obtained. This approximation will not significantly perturb the results because it may be readily shown that the heat transfer from the surfaces is small in comparison to that which is conducted. This assumption therefore will be incorporated into the analysis since it will not affect the results.

Now consider a small section of thermoelement:



The following energy balance will describe the section behavior:

$$\begin{bmatrix}\text{heat flow rate}\\ \text{into section at}\\ n-1/2\end{bmatrix} - \begin{bmatrix}\text{heat flow rate}\\ \text{out of section}\\ \text{at } n+1/2\end{bmatrix} + \begin{bmatrix}\text{rate of heat}\\ \text{generation}\\ \text{between } n-1/2\\ \text{and } n+1/2\end{bmatrix} =$$

$$\begin{bmatrix}\text{rate of heat accumulation}\end{bmatrix} \qquad \text{(II-1)}$$

The rate at which heat flows into the section is:

$$-k\ (t)_{n-1/2}\ A\ \left(\frac{\partial\ T\ (t)}{\partial\ Z}\right)_{n-1/2} \qquad \text{(II-2)}$$

where it is assumed that there is no capacitance or impedance effect, and :

k	=	thermal conductivity
t	=	time
A	=	cross sectional area normal to the temperature gradient
T	=	temperature
Z	=	position

The rate at which heat flows out of the section is:

$$-k(t)_{n+1/2}\, A \left(\frac{\partial T(t)}{\partial Z}\right)_{n+1/2} \qquad \text{(II-3)}$$

The rate of heat generation is:

$$\left[I(t)\right]^2 R(t)_n + \Upsilon(t)_n\, I(t) \left(\frac{\partial T(t)}{\partial Z}\right)_n \Delta Z \qquad \text{(II-4)}$$

where:

I	=	electrical current
R	=	electrical resistance of the nth segment
Υ	=	Thomson coefficient

and the first term represents the Joule heating, the second the Thomson effect.

A fictitious heat generation term can be used to represent the heat interchange between the thermoelement and its "surroundings". This term is:

$$-F(t)\, P\, \Delta Z \left[T(t)_n^{\,4} - \overline{T_s(t)^4}\right] \qquad \text{(II-5)}$$

where:

F	=	view factor
P	=	perimeter of thermoelement measured on a plane normal to the Z-axis
$\overline{T_s^4}$	=	surrounding temperature to the fourth power averaged

The rate of heat accumulation is:

$$C_p(t)_n\, \rho(t)_n\, A\, \Delta Z\, \frac{\partial T(t)_n}{\partial t} \qquad \text{(II-6)}$$

where:

C_p	=	heat capacity
ρ	=	density

Substituting these into Equation (II-1) yields:

$$-k(t)_{n-1/2} A \left(\frac{\partial T(t)}{\partial Z}\right)_{n-1/2} + k(t)_{n+1/2} A \left(\frac{\partial T(t)}{\partial Z}\right)_{n+1/2} +$$

$$+ \left[I(t)\right]^2 R(t)_n + \tau(t)_n I(t) \left(\frac{\partial T(t)}{\partial Z}\right)_n \Delta Z -$$

$$- F(t) P \Delta Z \left[T(t)_n^4 - \overline{T_s(t)^4}\right] = C_p(t)_n \rho(t)_n A \Delta Z \frac{\partial T(t)_n}{\partial t} \qquad \text{(II-7)}$$

The internal electrical resistance can be written:

$$R(t)_n = \frac{\rho(t)_n \Delta Z}{A} \qquad \text{(II-8)}$$

where ρ = electrical resistivity. Substituting this and taking the limit as ΔZ approaches zero results in:

$$\frac{\partial}{\partial Z}\left[k(Z,t)\frac{\partial T(Z,t)}{\partial Z}\right] + \frac{\rho(Z,t)\left[I(t)\right]^2}{A^2} - \frac{F(t)P}{A}$$

$$\left[T(Z,t)^4 - \overline{T_s(t)^4}\right] + \left(\frac{\tau(Z,t)\,I(t)}{A}\right)\frac{\partial T(Z,t)}{\partial Z} =$$

$$= C_p(Z,t)\,\rho(Z,t)\frac{\partial T(Z,t)}{\partial t} \qquad \text{(II-9)}$$

The physical properties are actually known as functions of T (Z, t). Further, the Thomson coefficient can be expressed as:

$$\tau(Z,t) = -T(Z,t)\frac{\partial \alpha(Z,t)}{\partial T(Z,t)} \qquad \text{(II-10)}$$

where α = Seebeck coefficient.

Making use of these, the differential equation becomes:

$$\frac{\partial}{\partial Z}\left[k(T)\frac{\partial T(Z,t)}{\partial Z}\right] + \frac{\rho(T)\left[I(t)\right]^2}{A^2} -$$

$$-\frac{F(t)P}{A}\left[T(Z,t)^4 - \overline{T_s(t)^4}\right] - \left[\frac{T(Z,T)\,I(t)}{A}\right]\frac{\partial \alpha(T)}{\partial Z} =$$

$$= C_p (T) \rho (T) \frac{\partial T (Z, t)}{\partial t} \qquad \text{(II-11)}$$

where the subscripts on the T have been left out if T is used as a subscript, but it should be realized that the variation still exists.

There is little chance of solving Equation (II-11) in a useful analytic manner, so we re-write it in the difference form:

$$\left[k (T) \frac{\partial T (t)}{\partial Z} \right]_{n + 1/2} - \left[k (t) \frac{\partial T (t)}{\partial Z} \right]_{n - 1/2} +$$

$$+ \frac{f (t)_n \left[I (t) \right]^2 \Delta Z}{A^2} - \frac{T (t)_n I (t)}{A} \left[\alpha (t)_{n + 1/2} - \right.$$

$$\left. - \alpha (t)_{n - 1/2} \right] - \frac{F (t) P}{A} \left[T (t)_n^{\,4} - \overline{T_s (t)^4} \right] =$$

$$= C_p (t)_n \rho (t)_n \frac{d T (t)_n \Delta Z}{dt} \qquad \text{(II-12)}$$

$$k (t)_{n + 1/2} \left[\frac{T (t)_{n + 1} - T (t)_n}{\Delta Z} \right] - k (t)_{n - 1/2} \left[\frac{T (t)_n - T (t)_{n - 1}}{\Delta Z} \right] +$$

$$+ \frac{f (t)_n \left[I (t) \right]^2 \Delta Z}{A^2} - \frac{T (t)_n I (t)}{A} \left[\alpha (t)_{n + 1/2} - \alpha (t)_{n - 1/2} \right] -$$

$$- \frac{F (t) P}{A} \left[T (t)_n^{\,4} - \overline{T_s (t)^4} \right] = C_p (t)_n \rho (t)_n \frac{d T (t)_n \Delta Z}{dt} \qquad \text{(II-13)}$$

In general:

$$\wp_{n + 1/2} \approx \frac{\wp_{n + 1} + \wp_n}{2} \qquad \text{(II-14)}$$

where $\wp$ is some property, so that:

$$\frac{\left[k(t)_{n+1} + k(t)_n\right]\left[T(t)_{n+1} - T(t)_n\right]}{2\,\Delta Z} -$$

$$- \frac{\left[k(t)_n + k(t)_{n-1}\right]\left[T(t)_n - T(t)_{n-1}\right]}{2\,\Delta Z} + \frac{\rho(t)_n \left[I(t)\right]^2 \Delta Z}{A^2} -$$

$$- \frac{T(t)_n\, I(t)}{2\,A}\left[\alpha(t)_{n+1} - \alpha(t)_{n-1}\right] -$$

$$- \frac{F(t)\,P}{A}\left[T(t)_n^{\,4} - \overline{T_s(t)^4}\right] = C_p(t)_n\, \rho(t)_n \frac{\partial T(t)_n\, \Delta Z}{\partial t} \qquad \text{(II-15)}$$

which can be rewritten as:

$$T_{n,\,j+1} = T_{n,\,j} + \frac{\Delta t}{C_{pn,\,j}\,\rho_{n,\,j}\,\Delta Z}\left\{\frac{k_{n+1,\,j} + k_{n,\,j}}{2\,\Delta Z}\right.$$

$$\left[T_{n+1,\,j} - T_{n,\,j}\right] - \frac{k_{n,\,j} + k_{n-1,\,j}}{2\,\Delta Z}\left[T_{n,\,j} - T_{n-1,\,j}\right] +$$

$$+ \frac{\rho_{n,\,j}\, I_j^{\,2}\,\Delta Z}{A^2} - \frac{T_{n,\,j}\, I_j}{2\,A}\left[\alpha_{n+1,\,j} - \alpha_{n-1,\,j}\right] -$$

$$\left. - \frac{F_j\,P}{A}\left[T_{n,\,j}^{\,4} - \overline{T_{s\,j}^{\,4}}\right]\right\} \qquad \text{(II-16)}$$

where the time derivative has been changed to a forward finite difference approximation, indicated by the j subscript. Note that the only j + 1 term is on the left side of the equation, which greatly simplifies calculations.

This equation describes the thermoelectric performance within the thermoelement but is not applicable to the node at each end. Minor modifications allow this equation to also be applied at the end nodes. For example, at the hot end the n - 1 node is meaningless. However, the temperature represented by the n - 1 node may be represented approximately by the adjoining node in the absorber to which the thermoelement is attached. The Seebeck coefficient for the n - 1 node may be treated as the Seebeck coefficient for the n^{th} node for this one point. This introduces negligible error because the boundary condition represented by the Peltier cooling is introduced separately as will be explained later. One of the thermal conductivity terms must also be eliminated in a similar manner. This introduces very small error in the result since essentially a numberical average is being obtained. The same type of changes can be made at the other end of the thermoelement where it contacts the radiator. In this case, the n + 1 node is meaningless, and is treated in a similar way.

The portion of the radiator and of the absorber which contacts the ends of the thermoelement is assumed to be a uniform temperature region with respect to position. This assumption is consistent with the previously stated assumption that there is no temperature gradient in a plane perpendicular to the thermoelement axis. It introduces only a very small error in the results because the volume involved is small and the temperature gradients over such a small volume will be minimal. The boundary conditions between the absorber and the thermoelement or between the radiator and the thermoelement are introduced during the treatment of this node. The rate at which heat is removed from the portion of the absorber in contact with the thermoelement can be described by:

$$Q_i(t) = -k(t)_1 A \left[\frac{\partial T(Z, t)}{\partial Z}\right]_{Z=0} + \alpha(t)_1 I(t) T(t)_1 - \left[I(t)\right]^2 R_{ci}(t) \qquad \text{(II-17)}$$

where the first term is the heat conducted away by the thermoelement, the second is the Peltier cooling, the third is the Joule heating term at the contact, and:

Q_i	=	heat flow into thermoelement
R_{ci}	=	contact resistance at the hot end

The subscript one refers to the first node in the thermoelement (the hot end). The heat flow rate out of the cold end is:

$$Q_o(t) = -k(t)_N A \left[\frac{\partial T(Z, t)}{\partial Z}\right]_{Z=\mathcal{Z}} + \alpha(t)_N I(t) T(t)_N + \left[I(t)\right]^2 R_{co}(t) \qquad \text{(II-18)}$$

where the subscripts N and o refer to the heat out (cold) end and $0 \leq Z \leq \mathcal{Z}$.

The derivative portions of these equations represent the change of temperature with respect to position within the thermoelement as the thermoelement end is approached from the inside. These terms may be immediately written in difference form and can therefore be readily evaluated using the previously presented difference equations. It is assumed that the contact resistances are known as functions of temperature from experimental data. Since the equations can be used to compute temperature, the contact resistances then can also be computed. The same is true of the other terms which appear in these equations. Again, the attractiveness of a forward difference technique is apparent because it is not necessary to iterate each time during the computation process. Were some other difference technique to be used, either iterative techniques or complicated approaches would be required which could be more time consuming. (The numerical approach will be expanded in later sections.)

The previously described equations are sufficient to enable thermoelectric performance to be calculated provided the electrical characteristics have been determined. However, the electrical characteristics depend upon the thermoelement temperatures and behavior. Consequently, these characteristics must be incorporated into the analysis so that a simultaneous solution may be obtained. To accomplsih this, we first consider the following equation which gives the total internal resistance of a thermoelement:

$$R_i(t) = \int_0^Z \frac{\rho(Z, t)}{A} \, dZ \tag{II-19}$$

The open circuit voltage is:

$$e(t) = \int_{T(t)_N}^{T(t)_1} \alpha(T) \, dT \tag{II-20}$$

These two equations represent the behavior for a single thermoelement. In practice, the indicated operations will be performed for an N element and a P element, and the resistances and the voltages will then be added to obtain the behavior for a complete couple. Multiplication of these results by the number of couples immediately yields the overall characteristics of the complete system. The following equation immediately results from incorporating these answers into an analysis of the system circuit including a load resistance:

$$e_N(t) + e_P(t) - I(t)\left[\frac{R_e(t)}{M} + R_t(t)\right] = 0 \tag{II-21}$$

where:

N refers to the negative element

P refers to the positive element

M = number of couples

R_e = external or load resistance

R_t = total internal resistance per couple

$= R_{iN}(t) + R_{iP}(t) + R_{CCN}(t) + R_{CCP}(t) + R_{CHN}(t) + R_{CHP}(t)$

R_{CC} = cold end contact resistance (including radiator)

R_{CH} = hot end contact resistance (including absorber)

The output power is now:

$$P(t) = I(t)^2 R_e(t) \tag{II-22}$$

The efficiency of the system can be defined in two ways:

$$\eta = \frac{\text{power delivered by thermoelement}}{\text{heat absorbed by thermoelement}} = \text{thermoelectric efficiency} \tag{II-23}$$

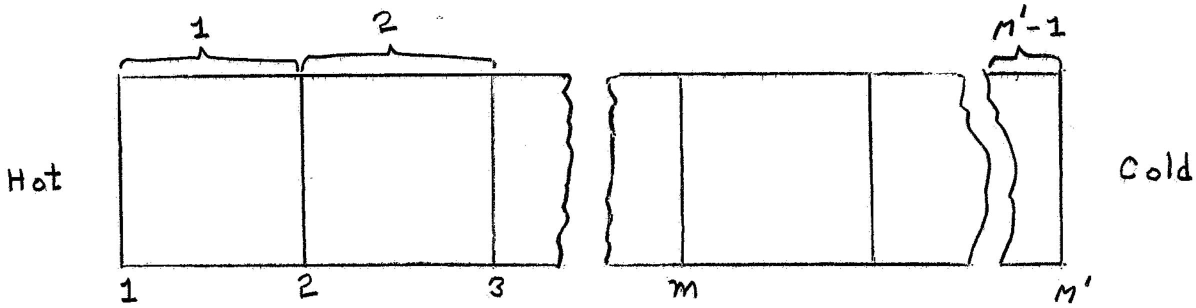
$$\eta = \frac{\text{useful power delivered to load}}{\text{heat incident on absorber}} = \text{overall efficiency} \tag{II-24}$$

It is interesting to note that the latter expression will become infinite when the solar heat is cut off, but the generator continues to function for a short time because of stored thermal energy.

To obtain a mean efficiency for any particular situation, it is only necessary to integrate over the time span of interest.

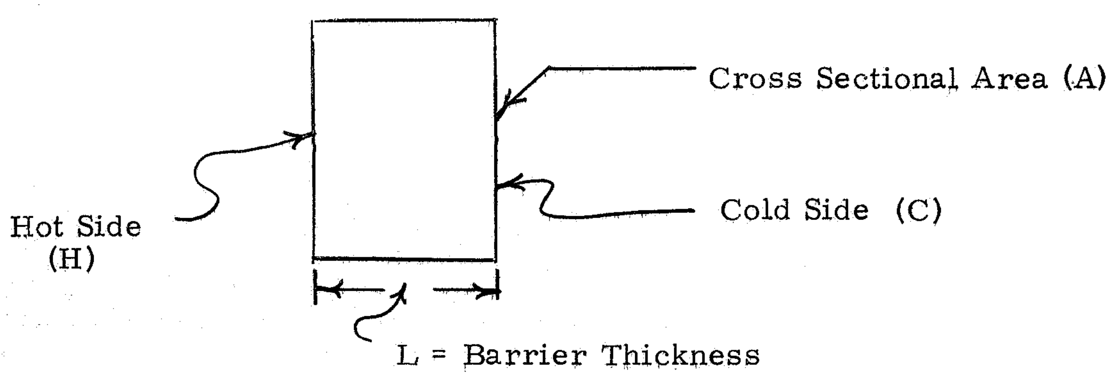
The same technique can be extended to a segmented thermoelement. However, it is necessary to include the effects of the discontinuity between the segments. These effects include contact resistances, inert material caused temperature drops, barrier losses, etc. Practical considerations of this type could cause an actual loss in switching to a segmented system as compared to a predicted gain for an ideal situation.

To gain further insight into the behavior of segmented systems, consider the following thermoelement leg:



The heat absorbed by this element at the hot end can be obtained by applying Equation (II-17) at this point. The heat given up at the cold end of segment 1 can be immediately determined by using Equation (II-18). The behavior within segment 1 is described by Equation (II-16). The same statements can be made for each of the M' - 1 segments of which the thermoelement is composed.

The use of Equations (II-17) and (II-18) automatically includes the effect of contact resistance between the segments. In reality, a temperature drop could occur in a barrier material between the segments. Since this material will be thin, its behavior can be described by considering:



The rate of heat flow entering the hot side is the heat given up by the thermoelement to the left. The heat flow rate out is that absorbed by the thermoelement on the right. These terms are already described by Equations (II-17) and (II-18). Therefore, if T (t) is the average temperature of the barrier we may write:

$$\begin{bmatrix}\text{rate of} \\ \text{heat in} \\ (Q_i(t))\end{bmatrix} - \begin{bmatrix}\text{rate of} \\ \text{heat out} \\ (Q_o(t))\end{bmatrix} = \begin{bmatrix}\text{rate of} \\ \text{heat} \\ \text{accumulation}\end{bmatrix} =$$

$$= \left[C_p \rho AL \frac{dT(t)}{dt}\right] \qquad \text{(II-25)}$$

The temperature drop across the barrier is approximately:

$$\frac{Q_i(t) + Q_o(t)}{2} \approx \frac{kA}{L}\left[T_i(t) - T_o(t)\right] \qquad \text{(II-26)}$$

and further:

$$T(t) \approx \frac{T_i(t) + T_o(t)}{2} \qquad \text{(II-27)}$$

so that the barrier behavior is now completely described.

If the barrier equations for each barrier (between segments) are combined with the equations and boundary conditions applicable to the segments, a set of difference equations is obtained similar to the set obtained for a non-segmented thermoelement, the basic difference being in the coefficients. Once these have been determined, the thermoelement points may be handled simultaneously with all of the other mesh points.

The electrical output and efficiency for the segmented system can now be obtained as easily as for the non-segmented system. The total electrical output is the summation of the electrical outputs of each segment minus the losses already discussed (with the additional assumption that the electrical resistance of the barriers is included in the contact resistance associated with those barriers). The efficiency is the net total electrical output divided by the heat into the hot end of each segment.

B. Absorber and Radiator Analysis

The behavior of the radiator can be handled by identical equations as utilized to analyze the absorber provided proper care is taken in setting up the analysis. Consequently, this discussion will be limited to a consideration of basically the absorber. All assumptions and considerations which are made will be in sufficient depth that the applicable equations may be readily utilized for the radiator as well.

The absorber acts as a heat transfer fin in that it receives radiant energy principally from the sun and transmits a major portion of this energy to the thermoelement hot junction by conduction. The remaining energy is either reradiated from the surface or is radiated directly to the radiator. Since one objective in the design of this system is to minimize weight, it is readily apparent that the absorber thickness must be made as thin as possible consistent with good heat transfer characteristics. Exactly what comprises "good heat transfer characteristics" cannot be determined without a detailed investigation. Investigations performed prior to the analysis reported herein have not considered the temperature gradients which result in the absorber and have therefore been unable to fully evaluate the performance characteristics.

Since the absorber is thin relative to its other dimensions, it may be treated as having a negligible temperature gradient perpendicular to the surface. This reduces the problem to a two-dimensional conduction analysis in Cartesian coordinates since the absorber is a rectangular structure. However, the nature of the reradiation terms from the surface as well as the complexities of the radiant energy interchange effects result in a mathematical representation which cannot be handled using analytic techniques.

The absorber receives heat principally by radiation from the sun, although energy interchange will also result from the albedo (planet reflected solar radiation), from radiation emitted directly by planets which is intercepted by the collector and by interchange between the absorber and the radiator. Although the absorber input may be quite accurately calculated using only the direct solar radiation, the behavior of the radiator can be strongly effected by the surroundings it "sees". Since these surroundings must be incorporated into the radiator analysis, they will also be incorporated into the absorber analysis which will, of course, increase the accuracy of the absorber calculation.

The calculation of the solar energy intercepted by the absorber is relatively straightforward from the standpoint that the flux is relatively well-known and it is only necessary to determine the orientation of the absorber with respect to the incoming solar flux. Knowledge of the relative absorptivity to solar radiation immediately enables one to compute the fraction of the impinging energy which is absorbed. Since the relative absorptivity can be a function of temperature, this dependency will be an allowed variable.

Powers* states that the calculation of "earthshine" is considerably less precise. He further states that fundamental assumptions are normally required to reduce the complexity of the computation. These normally consider that the earth is a diffusely emitting black body with a uniform surface temperature of $450^{\circ}R$. When this assumption has been made, it is immediately possible to compute the direct energy input impinging upon the

---

* Powers, Edward I., "Thermal Radiation to a Flat Surface Rotating About an Arbitrary Axis in an Eliptical Earth Orbit: Application to Spin-Stabilized Satellites", National Aeronautics and Space Administration, NASA TN D-2147, April 1964.

surface of the absorber from all portions of the earth which are "seen" from the absorber. Similar considerations are applicable to planets other than the earth.

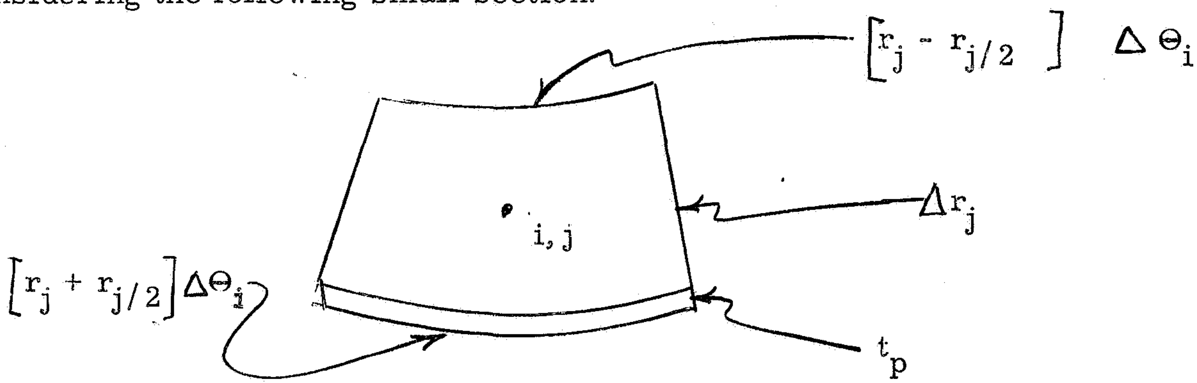
It is immediately apparent that calculation of the effect of albedo requires a similar treatment. Powers in this case assumes that the earth can be represented as a diffusely reflecting sphere with the source intensity represented as a function of the absorber location relative to that portion of the earth which is lit by the sun.

Heat will be reradiated from the surface of the absorber. This may be immediately determined if the temperature is being computed by utilizing the relative emissivity which can be represented as a function of temperature.

Energy interchange will also occur between the absorber and the radiator. Although this calculation may be readily made between two parallel flat plates if the surfaces are at uniform temperatures, the problem becomes more complex if temperature distributions occur, as is the case under consideration here. Since the surfaces of the absorber and radiator which "see" each other will be highly polished to minimize the energy interchange, the actual quantity of heat transmitted between the two surfaces will be small in comparison to the other heat transfer effects. Consequently, if some error occurs in this portion of the analysis, its effect on the overall performance will be minimized. As a result, the assumption will be made that the collector and the radiator can each be represented in terms of the fourth power of temperature integrated over the surface of the plate in question and hence averaged. This integration process is straightforward once the temperature distribution has been established. Immediately, the energy interchange between the two surfaces may now be treated as though the surfaces had a uniform temperature. The overall effect of this assumption is a slight perturbation in the temperature distribution on the surfaces. At the edges which are furthest removed from the thermoelement, the absorber and radiator will normally have the largest temperature difference, and the energy interchange at this point will be slightly greater than computed with the assumption. Conversely, the opposite effect will occur as the portion of the surface in question is considered as approaching the thermoelement.

The basic solar flat plate configuration as presently constructed and analyzed herein can be represented by a rectangular geometry. This would immediately imply that the basic analysis of the absorber should also be based on rectangular geometry. Indeed, initial approaches to the analysis of this problem were made utilizing this assumption. However, a consideration of the heat flux and hence the nature of the temperature gradients within the absorber resulted in a change from this geometry to the cylindrical coordinate system. This apparently contradictory decision was reached from consideration of two different standpoints. First, most of the energy which is absorbed by the absorber is transferred by conduction to the thermoelement. The heat fluxes in the vicinity of the hot end of the thermoelement are therefore high, with large temperature gradients, and these heat fluxes decrease as the distance from the hot end of the thermoelement is increased. This means that a numerical representation of the behavior of the absorber should be based upon selection of small volumes or nodes which are located closely together. Since the temperature gradients become less as the distance from the thermoelement is increased, nodes which are located further away may be utilized to represent larger volumes with a similar accuracy. Secondly, it was desired that the computer program should be as easily used as possible. This essentially meant that the computer program should be caused to generate its own nodal arrangement based upon a minimum of input data. Both of these considerations immediately imply a cylindrical coordinate system as one which most readily allows the desired node arrangements to be attained. This assumption is discussed further and in more detail later in this report (see also Appendix A).

A heat balance may be taken over a small section of the absorber by considering the following small section:



The following heat balance can be written:

$$\text{heating rate} - \text{rate of heat accumulation in segment} - \text{radiant heat loss rate} = \text{rate of heat conduction away from segment} \quad \text{(II-28)}$$

The total heat flux per unit area absorbed by the exposed surface is given by:

$$Q_{i,j}(t) = S\alpha(\delta, T_{i,j})\beta(\delta)\cos(\delta(t)) + \sigma F_1(T_{i,j}, t)\,Tp(t)^4 +$$

$$+ S\,R\,F_2(T_{i,j}, t) \quad \text{(II-29)}$$

where:

Q	=	absorbed heat flux
S	=	solar constant
α	=	absorptivity (for solar radiation)

δ	=	incident angle for solar radiation measured from a line perpendicular to the plate surface
$T_{i,j}$	=	temperature at position, i, j
β	=	$\left.\begin{array}{l} 1 \text{ if } 0 \leq \delta < \pi/2 \\ 0 \text{ otherwise} \end{array}\right\}$ for collector
β	=	$\left.\begin{array}{l} 0 \text{ if } 0 \leq \delta < \pi/2 \\ 1 \text{ otherwise} \end{array}\right\}$ for radiator
t	=	time
σ	=	Stefan-Boltzmann constant
F_1	=	radiation view factor for heat transfer to plate from a planet
T_p	=	planet absolute temperature
R	=	reflectivity constant of planet (albedo)
F_2	=	radiation view factor for reflected (solar) heat transfer to plate from planet

Note that Q is not independent of location on the plate since the relative absorptivities are functions of temperature. The heat flow associated with the surface is referred to in the following way to simplify the algebra:

$$q = Q(T_{i,j}, t)\, r_j\, \Delta r\, \Delta \Theta \tag{II-30}$$

where:

q	=	the heat absorbed per unit time by the segment for this phenomenon

Heat will also be absorbed from the other plate (if we are considering the absorber, the other plate is the radiator). The net heat transferred can be described by:

$$q = \sigma F_{i,j}(T_{i,j}, T'_{i,j}) \left[T_{i,j}(t)^4 - \overline{T'(t)^4} \right] r_j\, \Delta r\, \Delta \Theta \tag{II-31}$$

where:

$F_{i,j}$	=	view factor
$T_{i,j}$	=	temperature at (r, Θ)
$\overline{T'^4}$	=	effective temperature of other plate taken to the fourth power and averaged

Heat will be radiated from the exposed surface according to:

$$q = \sigma \epsilon_o (T_{i,j}) \Delta r \, r_j \, \Delta \Theta \; T_{i,j}(t)^4 \tag{II-32}$$

where:

ϵ_o = relative emissivity of exposed surface

Heat conducted from the plate segment into other portions of the plate can be described by:

$$q = \sum \text{(rate of heat conduction out of segment through sides)} \tag{II-33}$$

which can be written:

$$q = - t_p \Big\{ k(T_{i,j-\frac{1}{2}}) \frac{\Delta \Theta}{\Delta r} \Big[r_j - \frac{\Delta r}{2} \Big] \Big[T_{i,j-1}(t) - T_{i,j}(t) \Big] + $$

$$+ k(T_{i,j+\frac{1}{2}}) \frac{\Delta \Theta}{\Delta r} \Big[r_j + \frac{\Delta r}{2} \Big] \Big[T_{i,j+1}(t) - T_{i,j}(t) \Big] + $$

$$+ \frac{k(T_{i+\frac{1}{2},j})}{r_j} \frac{\Delta r}{\Delta \Theta} \Big[T_{i+1,j}(t) - T_{i,j}(t) \Big] + \frac{k(T_{i-\frac{1}{2},j})}{r_j} \frac{\Delta r}{\Delta \Theta}$$

$$\Big[T_{i-1,j}(t) - T_{i,j}(t) \Big] \Big\} \tag{II-34}$$

Finally, heat will be accumulated within the segment due to a change in temperature with respect to time. This effect is:

$$q = C_p (T_{i,j}) \rho (T_{i,j}) t_p r_j \Delta \Theta \Delta r \frac{dT_{i,j}(T)}{dt} \tag{II-35}$$

All other effects are assumed negligible, as previously discussed.

These effects can be combined accordingly to yield:

$$C_p (T_{i,j}) \rho (T_{i,j}) t_p r_j \Delta \Theta \Delta r \frac{dT_{i,j}(t)}{dt} = Q(T_{i,j}, t) r_j \Delta r \Delta \Theta -$$

$$-\sigma F_{i,j} (T_{i,j}, T'_{i,j}) \Big[T_{i,j}(t)^4 - \overline{T'^4} \Big] r_j \Delta \Theta \Delta r + S_{i,j}(t) t_p \Delta r \, r_j \Delta \Theta -$$

$$
\begin{aligned}
&-\sigma \epsilon_o (T_{i,j}) \Delta r\, r_j\, \Delta\Theta\, T_{i,j}(t)^4 + t_p\, k\,(T_{i,\, j-\frac{1}{2}}) \frac{\Delta\Theta}{\Delta r} \\
&\left[r_j - \frac{\Delta r}{2}\right] \left[T_{i,\,j-1}(t) - T_{i,j}(t)\right] + k\,(T_{i,\,j+\frac{1}{2}}) \frac{\Delta\Theta}{\Delta r} \\
&\left[r_j + \frac{\Delta r}{2}\right] \left[T_{i,\,j+1}(t) - T_{i,j}(t)\right] + \frac{k\,(T_{i+\frac{1}{2},\, j})}{r_j} \frac{\Delta r}{\Delta\Theta} \\
&\left[T_{i+1,\,j}(t) - T_{i,j}(t)\right] + \frac{k\,(T_{i-\frac{1}{2},\, j})}{r_j} \frac{\Delta r}{\Delta\Theta} \Big[T_{i-1,\,j}(t) - \\
&- T_{i,j}(t)\Big]\Big\}
\end{aligned}
\tag{II-36}
$$

which may be transformed to the following differential equation by dividing by $r_j t_p \Delta\Theta \Delta r$ and allowing Δr and $\Delta\Theta$ to approach zero:

$$
\begin{aligned}
&C_p(T)\,\rho(T) \frac{\partial T(\Theta, r, t)}{\partial t} = \frac{Q(T,t)}{t_p} - \frac{\sigma}{t_p} F(\Theta, r, T, T') \\
&\left[T(\Theta, r, t)^4 - \overline{T'^4}\right] + S(\Theta, r, t) - \frac{\sigma}{t_p} \epsilon_o(T)\, T(\Theta, r, t)^4 + \\
&+ \frac{1}{r_j} \frac{\partial}{\partial r} \left\{ rk(T) \frac{\partial T(\Theta, r, t)}{\partial r} \right\} + \frac{1}{r_j^2} \frac{\partial}{\partial \Theta} \left\{ k(T) \frac{\partial T(\Theta, r, t)}{\partial \Theta} \right\}
\end{aligned}
\tag{II-37}
$$

which is the governing differential equation describing heat transfer in the absorber. Note that the direct functional dependencies are indicated. Hence, $F(\Theta, r, T, T')$ indicates that F is a function of the independent variables Θ and r, and of the dependent variables T and T'. F is not directly a function of time, although it will vary with time because T and T' would be completely written as $T(\Theta, r, t)$ and $T'(t)$.

The view factor F may be written as:

$$
F_{i,j} = \frac{1}{\dfrac{1}{\epsilon_o(T)} + \dfrac{1}{\epsilon'} - 1}
\tag{II-38}
$$

where ϵ' is the average relative emissivity of the absorber.

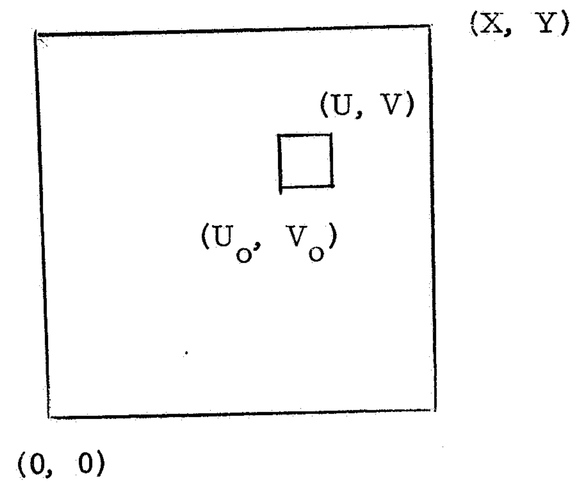
Similar equations for the radiator may be obtained by exchanging the T and T' terms and applying the other parameters properly.

The boundary conditions necessary to obtain solutions to the equations which represent the absorber or radiator essentially consist of the previously discussed interconnections between these surfaces and the thermoelements as well as the additional assumptions that no heat is lost from the edges of the array which are exposed to the surroundings. This latter assumption is relatively good since the internal portions of the array do not "see" space to a significant degree. The view factor for the internal portions to space is further decreased by the lip that is formed around the edge of the collector which serves as a stiffening member. By the same token, a portion of the heat that is collected by the absorber will be perturbed by the presence of this lip, although this has not been considered in the analysis since the effect will be small.

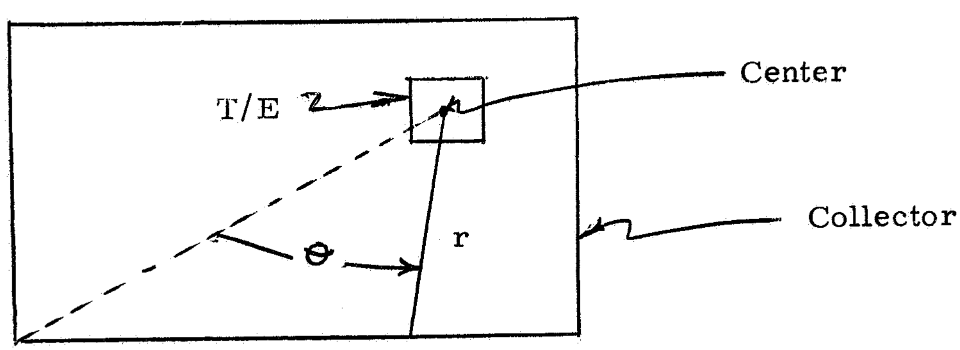
C. Mesh Set Up Technique

As previously discussed, heat conduction within either the absorber or the radiator may be described by taking a heat balance over a small representative section and then considering the equations which result as the section size is caused to approach zero. The resulting differential equation cannot be solved by present analytical techniques, and a numerical solution is therefore required. The solution selected basically consists of the expansion of the derivative terms through a Taylor series expansion with the rejection of higher order terms. This technique is commonly used for heat transfer calculations and yields excellent accuracy provided the increment sizes are properly selected. It further introduces relatively simple computation techniques which in turn speed up the solution through faster computation. A basic consideration is the selection of a suitable mesh arrangement which then forms the basis for the application of the differential equation.

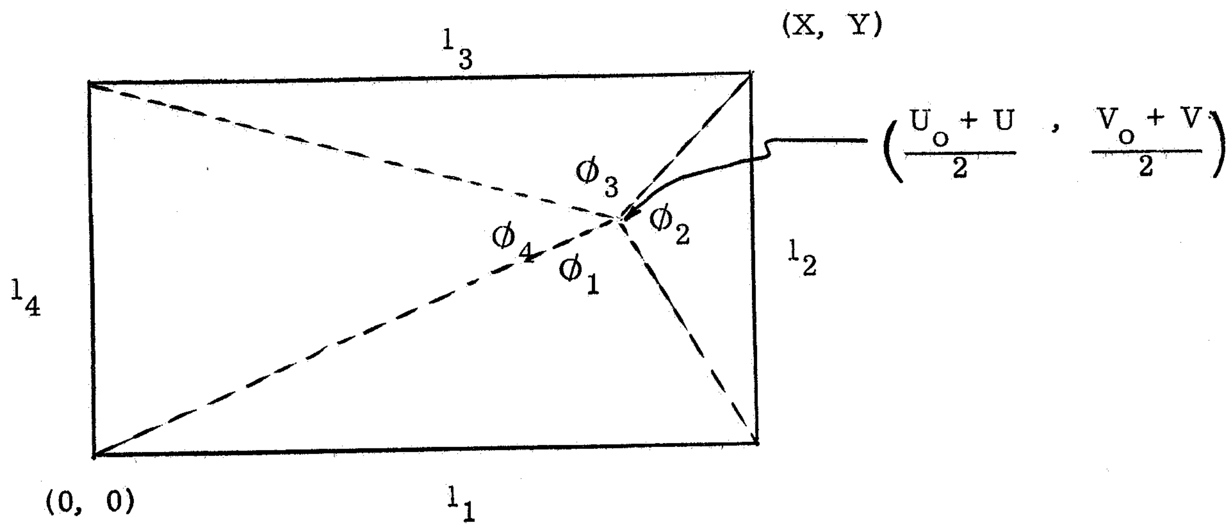
For reference purposes the absorber will be considered as oriented in a Cartesian (x, y) coordinate system with the origin located at the lower left corner. Specification of the plate size may be immediately made by defining the point (X, Y) as the coordinate of the upper right corner. The lower left corner of the thermoelement will be specified by (U_o, V_o), and the upper right corner will be specified by (U, V). This defines the overall dimensions of the absorber, as seen from the following sketch:



The calculations will be performed using radial geometry and selecting the radial increment, Δr, such that the controlling angular increment will approximately form a square. The controlling angular increment will be defined as the average value of $\Delta \Theta$ encountered in $0 \leq \Theta \leq 2\pi$:



To determine the $\Delta \Theta$'s we next consider:



where the l_i are the number of increments each of the i sides is to be divided into for calculation purposes. We can use the law of cosines to write:

$$\cos \phi_1 = \frac{4X^2 - (U_o + U)^2 - (2X - U_o - U)^2 - 2\,(V_o + V)^2}{-2\sqrt{(U_o + U)^2 + (V_o + V)^2}\sqrt{(2X - U_o - U)^2 + (V_o + V)^2}} \qquad \text{(II-39)}$$

$$\cos \phi_2 = \frac{4Y^2 - 2(2X - U_o - U)^2 - (V_o + V)^2 - (2Y - V_o - V)^2}{-2\sqrt{(2X - U_o - U)^2 + (V_o + V)^2}\sqrt{(2Y - V_o - V)^2 + (2X - U_o - U)^2}} \qquad \text{(II-40)}$$

$$\cos \phi_3 = \frac{4X^2 - 2\,(2Y - V_o - V)^2 - (2X - U_o - U)^2 - (U_o + U)^2}{-2\sqrt{(2Y - V_o - V)^2 + (2X - U_o - U)^2}\sqrt{(U_o + U)^2 + (2Y - V_o - V)^2}} \qquad \text{(II-41)}$$

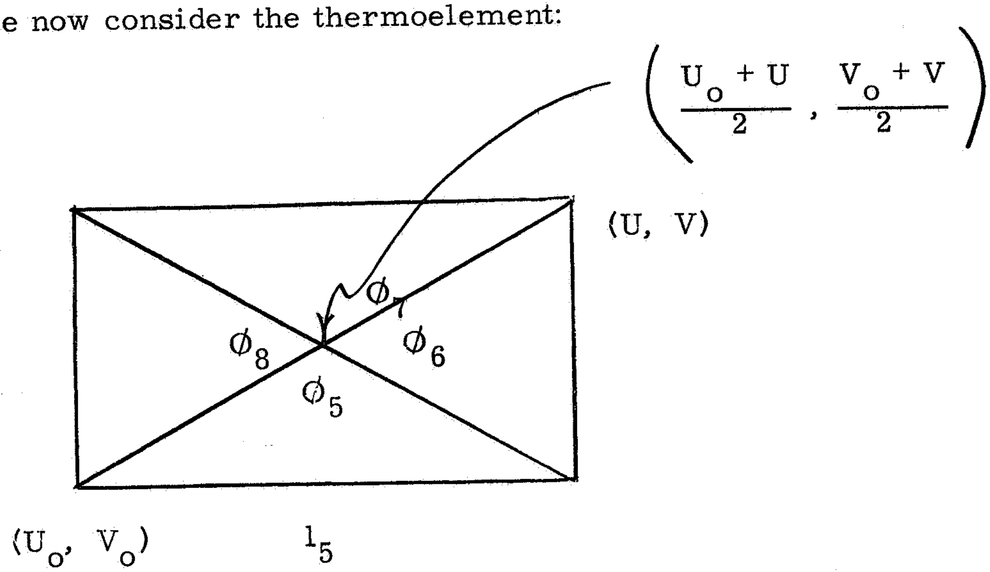
$$\phi_4 = 2\pi - \phi_1 - \phi_2 - \phi_3 \qquad \text{(II-42)}$$

Now define:

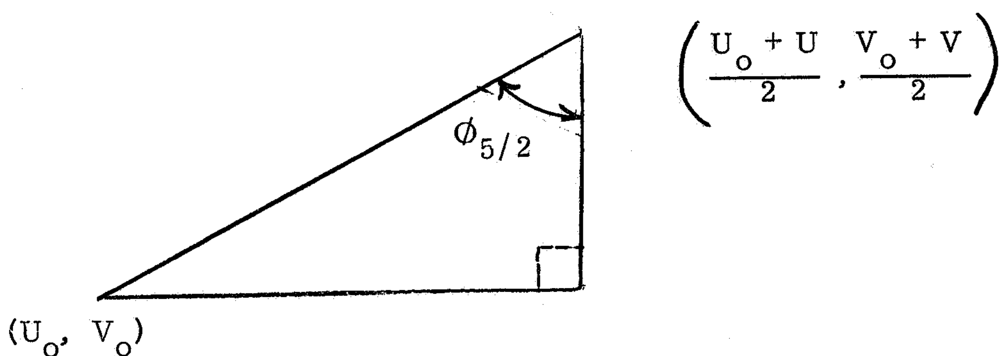
$$\Delta\, \Theta_i' = \frac{\phi_i}{l_i} \qquad 1 \leq i \leq 4 \qquad \text{(II-43)}$$

These are the $\Delta\, \Theta_i$ required by the l_i.

We now consider the thermoelement:



or, perhaps a little easier:



$$\tan \left(\frac{\phi_5}{2}\right) = \frac{U - U_o}{V - V_o} \tag{II-44}$$

Similarly:

$$\tan \left(\frac{\phi_6}{2}\right) = \frac{V - V_o}{U - U_o} \tag{II-45}$$

$$\phi_7 = \phi_5 \tag{II-46}$$

$$\phi_8 = \phi_6 \tag{II-47}$$

and:

$$\Delta \Theta_i' = \frac{\phi_i}{l_i} \qquad 5 \leq i \leq 8 \tag{II-48}$$

We now compare $\Delta\Theta_1'$ and $\Delta\Theta_5'$ and select the smaller of the two values to apply as $\overline{\Delta\Theta}_1$ for $0 \leq \Theta \leq \emptyset_1$. If $\Delta\Theta_5'$ is selected, a new l_1 corresponding to $\Delta\Theta_5'$ will be computed. The new l_1 will be truncated to the next larger integer if it is not already an integer, and the $\Delta\Theta_5'$ will be recomputed using the truncated l_1. This $\Delta\Theta_5'$ will become $\overline{\Delta\Theta}_1$.

Similar manipulations will be performed for $\emptyset_2$, $\emptyset_3$ and $\emptyset_4$.

Typical of the type of angular arrangement which could result from this specification is the geometry shown in Figure II-1. Note that the size of the rays on each side of line "A" are different. If carried to an extreme, this would mean that adjoining segments constructed on this angular arrangement could be of drastically different size. There would immediately result an error in the representation of the differential equation by node arrangements of this type. As a consequence, it is desirable that the variation from ray to ray be minimized. This can be accomplished through an iterative averaging technique which will result in the same number of rays/side, but will arrange the size somewhat differently. To accomplish this, we first define:

$$n_o = 0$$

$$n_1 = l_1$$

$$n_2 = n_1 + l_2 \tag{II-49}$$

$$n_3 = n_2 + l_3$$

$$n_4 = n_3 + l_4$$

$$\Delta\Theta_i = \overline{\Delta\Theta}_j \qquad 1 \leq j \leq 4, \; n_{j-1} + 1 \leq i \leq n_j \tag{II-50}$$

Now we select j = 1 and perform the following calculations:

$$\Delta\Theta_i = \frac{1}{2}\left[\Delta\Theta_{i+1} + \Delta\Theta_{i-1}\right] \tag{II-51}$$

for $i = n_{j-1} + 1$, then $n_{j-1} + 2$, $n_{j-1} + 3$, . . . , to n_j (inclusive).

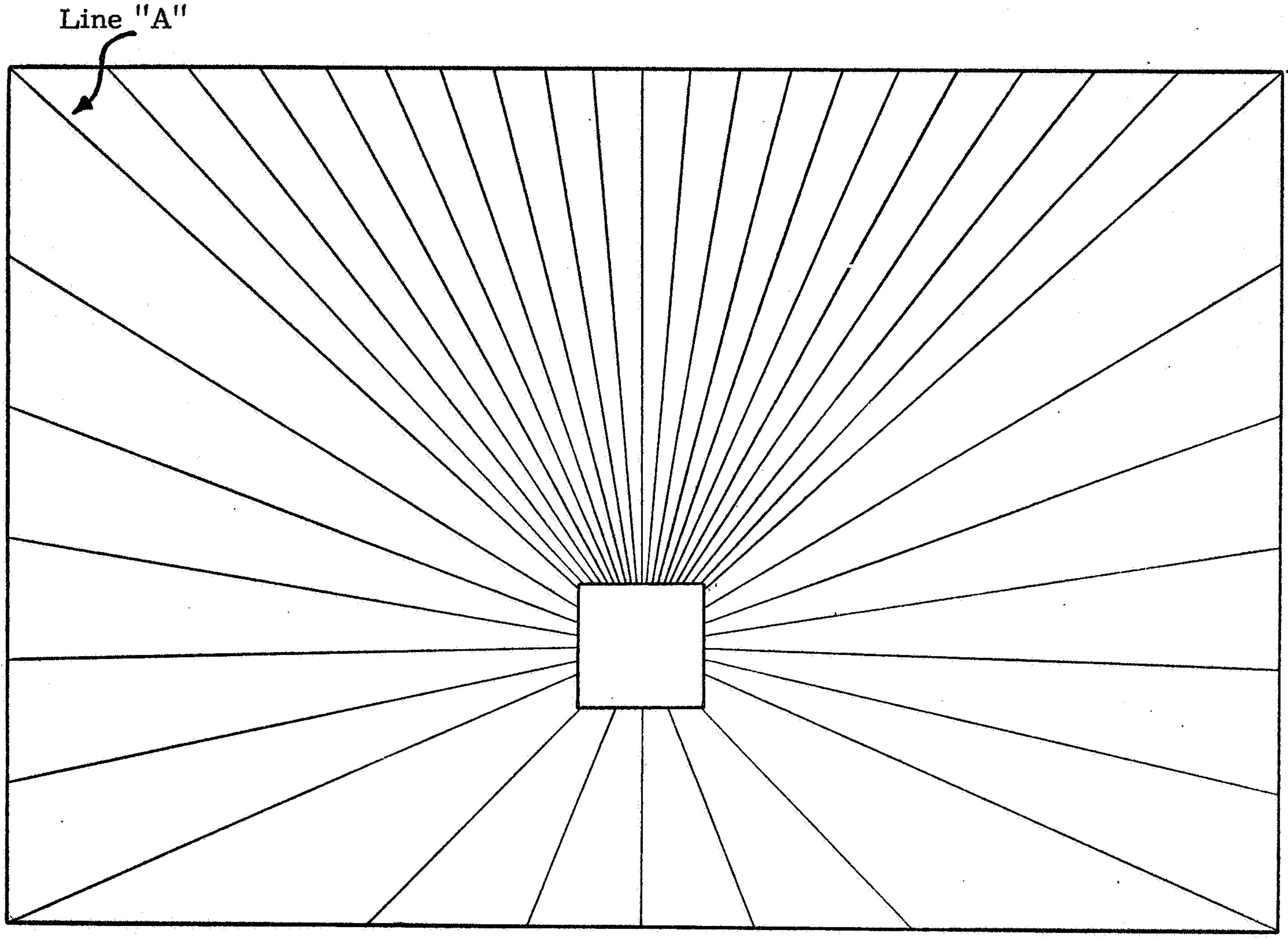


Figure II-1 **Non-Smoothed Angles**

A subscript zero will be taken to mean n_4, and a subscript $n_4 + 1$ to mean one. We now compute:

$$\phi = \sum_{i = n_{j-1}+1}^{n_j} (\Delta \Theta_i) \qquad \text{(II-52)}$$

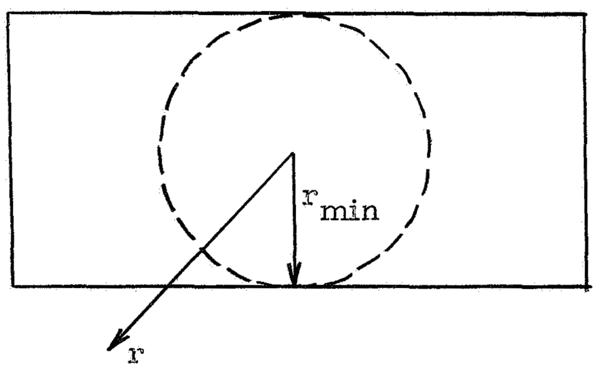
and we renormalize according to:

$$\Delta \Theta_i = \frac{\phi_j}{\phi} \Delta \Theta_i \qquad n_{j-1} + 1 \leq i \leq n_j \qquad \text{(II-53)}$$

where the meaning of Equation (II-53) is to compute the right hand side and then replace $\Delta \Theta_i$ by that value.

We now repeat Equations (II-51) through (II-53) for $j = 2$, $j = 3$, and $j = 4$. Then we repeat for $j = 1, 2, 3, 4$, and we continue the process until convergence has been obtained. This could typically result in the behavior illustrated in Figure II-2. Note that the variation of size from ray to ray has been smoothed, and no large discontinuities exist. The desired number of rays per side is still the same.

This process will set up the $\Delta \Theta_i$. We next must compute the r_j. To do this we first compute an initial radius as the minimum of $\frac{U - U_o}{2}$ and $\frac{V - V_o}{2}$. This is then the minimum value the radius can have and still be in the plate, but outside of the thermoelement.



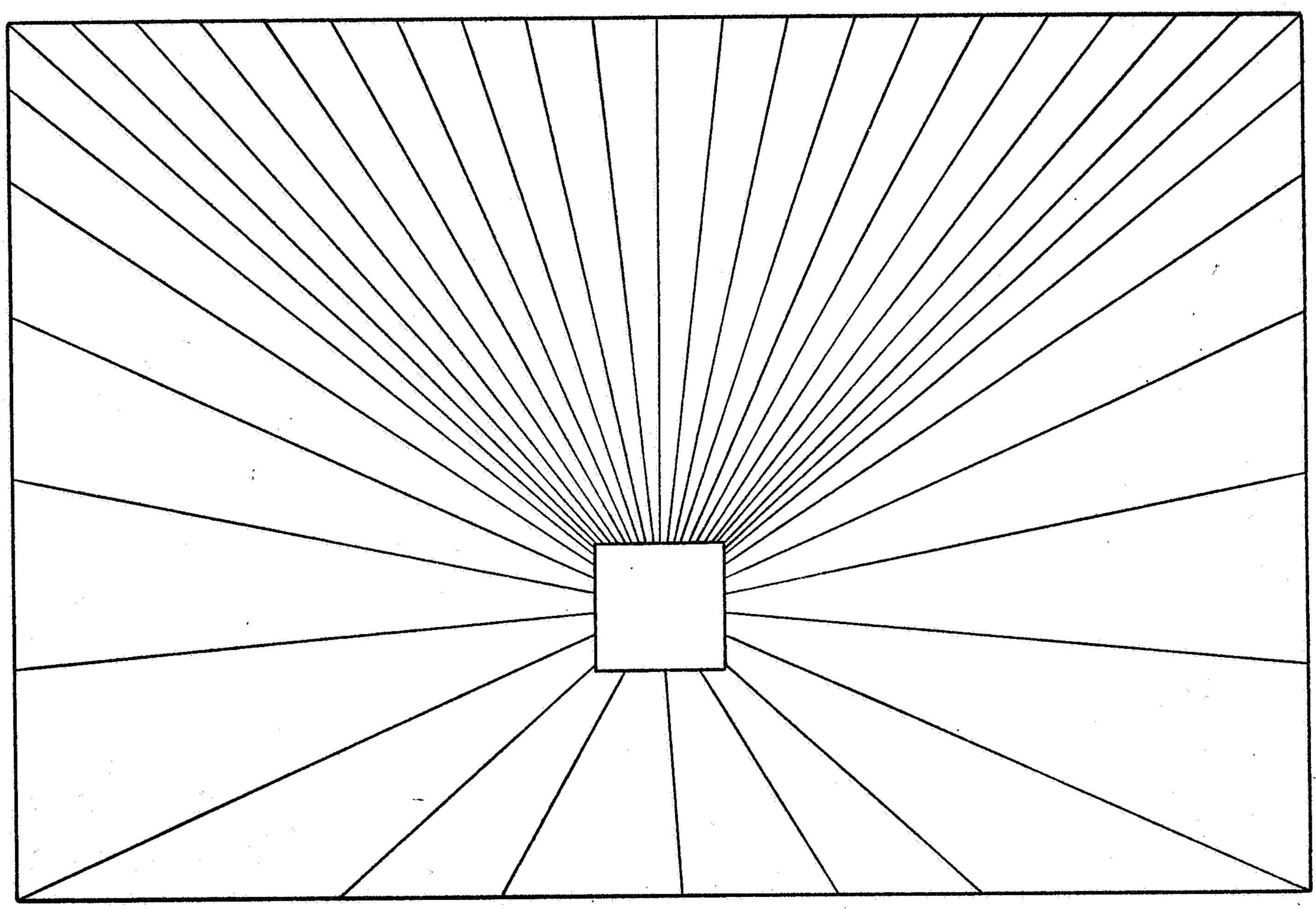


Figure II-2. **Illustration of "Smoothed" Angles**

We will base Δr_j on the average $\Delta \Theta_i$, Δr_j will be calculated by:

$$\Delta r_j = r_j \, \overline{\Delta \Theta} \tag{II-54}$$

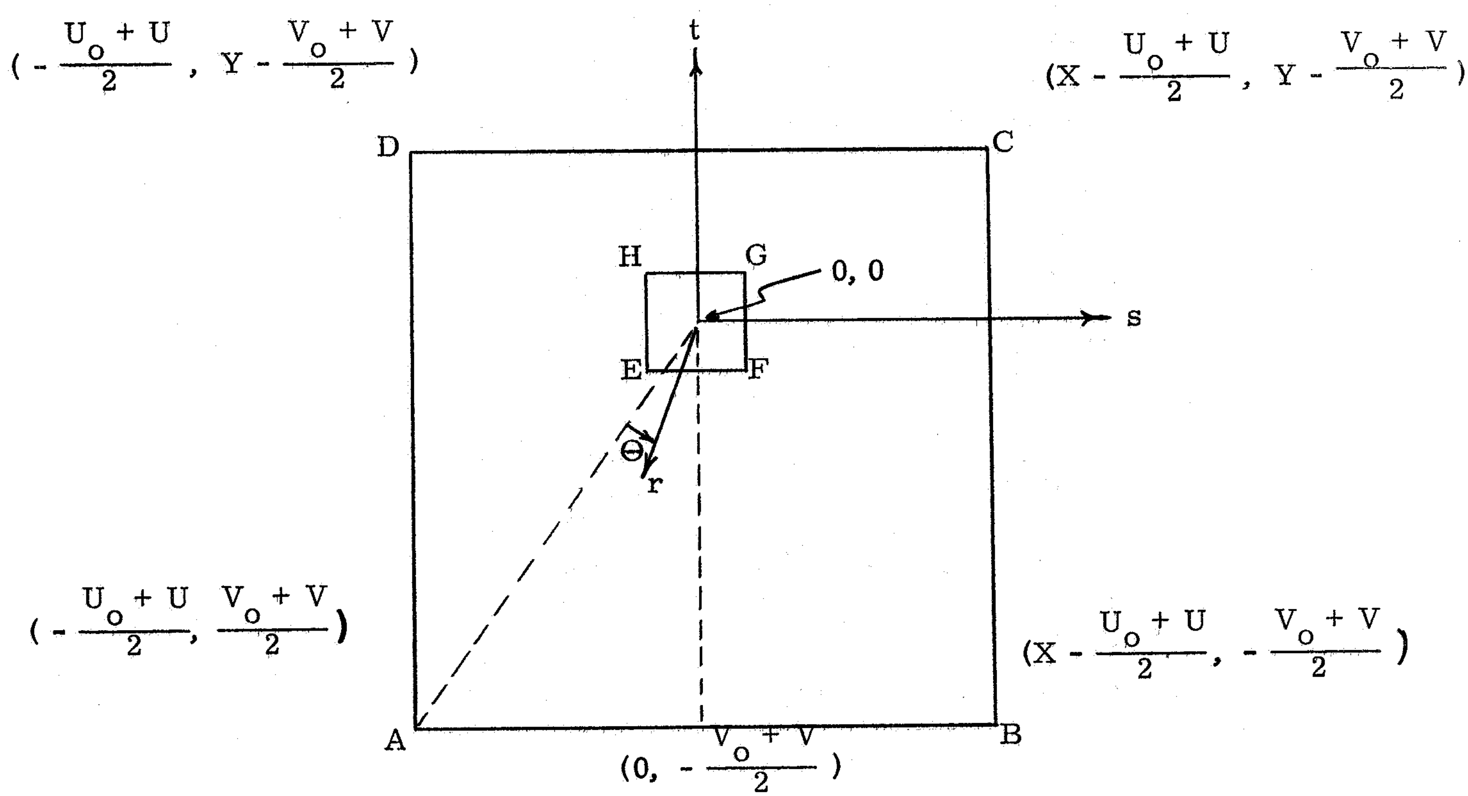
Thus, we see that Δr_j will increase as we move away from the thermoelement center. The initial Δr_j, will be calculated from r_{min}.

Then we can use:

$$r_{j+1} = r_j + \Delta r_j \tag{II-55}$$

to compute all of the other r, of course recomputing Δr_j each time.

To set up the mesh and take into account the thermoelement and the plate edges requires a knowledge of where they are located relative to the center of rotation. This may be accomplished by a transformation of coordinates to (r, Θ) geometry with the center at $x = \frac{U_o + U}{2}$, $y = \frac{V_o + V}{2}$:



where the coordinates are given in terms of the (s, t) system, a rectangular system with the same zero point as the (r, Θ) system and otherwise oriented parallel to the (x, y) system. In (r, Θ) geometry, the values for points A - D are obvious. To obtain E - H, we first note that when the radius vector is straight down:

$$\tan \Theta = \frac{U_o + U}{V_o + V} \tag{II-56}$$

Or:

$$\Theta = \tan^{-1} \left(\frac{U_o + U}{V_o + V} \right) \tag{II-57}$$

But at this point, we also have a known $\phi_5/2$ (See Equation (II-44)):

$$\frac{\phi_5}{2} = \tan^{-1} \left(\frac{U - U_o}{V - V_o} \right) \tag{II-58}$$

The value of Θ corresponding to E may be obtained by subtracting:

$$\Theta_E = \tan^{-1} \left(\frac{U_o + U}{V_o + V} \right) - \frac{\phi_5}{2} \tag{II-59}$$

But we will not be concerned here with negative Θ's. Hence we will stipulate that if Θ_E is negative as calculated, then corner E will be shifted to position F, F to G, etc. Immediately, corner F is located at $\Theta = \Theta_E + \phi_5$, etc. Since the center of the coordinate system is located in the center of thermoelement, the radii for the four corners are identical and may be obtained immediately. These data are summarized in Table II-1.

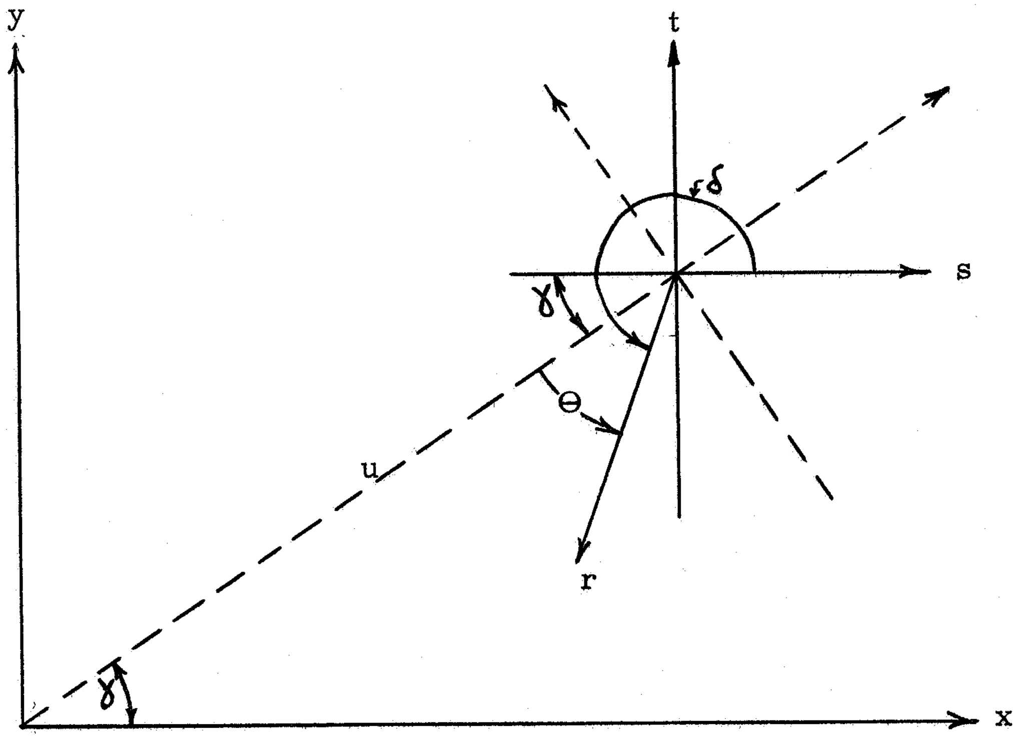
If (u, v) represents a rectangular coordinate system based upon the same zero as the polar coordinate system being considered here, then we may write:

$$u = r \cos \Theta \tag{II-60}$$

$$v = r \sin \Theta \tag{II-61}$$

TABLE II-1. POSITION OF CORNERS IN (r, Θ) GEOMETRY

Corners	Θ	r
A	0	$\frac{1}{2}\sqrt{(U_o + U)^2 + (V_o + V)^2}$
B	ϕ_1	$\sqrt{\left(X - \frac{U_o + U}{2}\right)^2 + \left(\frac{V_o + V}{2}\right)^2}$
C	$\phi_1 + \phi_2$	$\sqrt{\left(X - \frac{U_o + U}{2}\right)^2 + \left(Y - \frac{V_o + V}{2}\right)^2}$
D	$\phi_1 + \phi_2 + \phi_3$	$\sqrt{\left(\frac{U_o + U}{2}\right)^2 + \left(Y - \frac{V_o + V}{2}\right)^2}$
E	Θ_E	$\frac{1}{2}\sqrt{(U - U_o)^2 + (V - V_o)^2}$
F	$\Theta_E + \phi_5$	$\frac{1}{2}\sqrt{(U - U_o)^2 + (V - V_o)^2}$
G	$\Theta_E + \phi_5 + \phi_6$	$\frac{1}{2}\sqrt{(U - U_o)^2 + (V - V_o)^2}$
H	$\Theta_E + 2\phi_5 + \phi_6$	$\frac{1}{2}\sqrt{(U - U_o)^2 + (V - V_o)^2}$



and if (s, t) represents a rectangular coordinate system based at the same zero as the polar coordinate system, but rotated clockwise from the (u, v) system by $\pi + \gamma$, then:

$$\delta = \Theta + \gamma + \pi \tag{II-62}$$

but:

$$\tan \gamma = \frac{V_o + V}{U_o + U} \tag{II-63}$$

$$\gamma = \tan^{-1}\left(\frac{V_o + V}{U_o + U}\right) \tag{II-64}$$

and:

$$\delta = \Theta + \tan^{-1} \left(\frac{V_o + V}{U_o + U}\right) + \pi \tag{II-65}$$

Further:

$$s = r \cos \delta \tag{II-66}$$

$$t = r \sin \delta \tag{II-67}$$

Or:

$$s = r \cos \left[\Theta + \pi + \tan^{-1} \left(\frac{V_o + V}{U_o + U} \right) \right] \tag{II-68}$$

$$t = r \sin \left[\Theta + \pi + \tan^{-1} \left(\frac{V_o + V}{U_o + U} \right) \right] \tag{II-69}$$

We may also write:

$$t = y - \frac{V_o + V}{2} \tag{II-70}$$

$$s = x - \frac{U_o + U}{2} \tag{II-71}$$

So that:

$$x = r \cos \left[\Theta + \pi + \tan^{-1} \left(\frac{V_o + V}{U_o + U} \right) \right] + \frac{U_o + U}{2} \tag{II-72}$$

$$y = r \sin \left[\Theta + \pi + \tan^{-1} \left(\frac{V_o + V}{U_o + U} \right) \right] + \frac{V_o + V}{2} \tag{II-73}$$

But:

$$\sin (u + v) = \sin (u) \cos (v) + \cos (u) \sin (v) \tag{II-74}$$

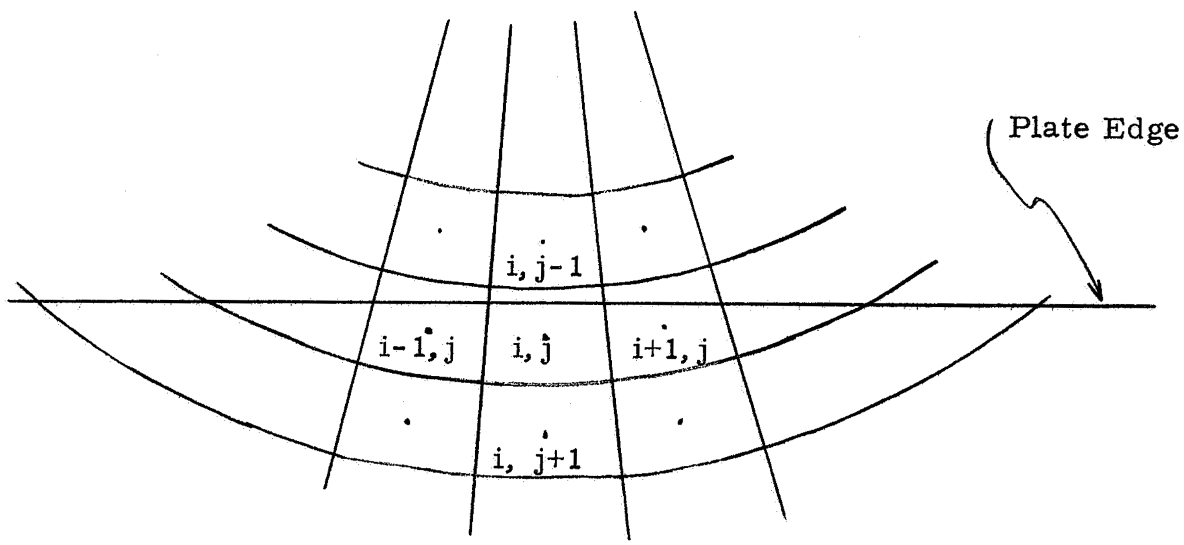
$$\cos (u + v) = \cos (u) \cos (v) - \sin (u) \sin (v) \tag{II-75}$$

and:

$$x = -r \left\{ \cos (\Theta) \cos \left(\tan^{-1} \left(\frac{V_o + V}{U_o + U} \right)\right) - \sin (\Theta) \sin \left(\tan^{-1} \left(\frac{V_o + V}{U_o + U} \right)\right) \right\} + \frac{U_o + U}{2} \tag{II-76}$$

$$y = -r \left\{ \sin(\Theta) \cos\left(\tan^{-1}\left(\frac{V_o + V}{U_o + U}\right)\right) + \right.$$
$$\left. + \cos(\Theta) \sin\left(\tan^{-1}\left(\frac{V_o + V}{U_o + U}\right)\right) \right\} \frac{V_o + V}{2} \qquad \text{(II-77)}$$

The plate edges will not fit into the mesh selected for analysis in a regular manner, and we must investigate these in more detail in order to generate a mesh which accurately represents the edge behavior as well as internal behavior. We first consider the following:



The plate edge is a straight line which, if horizontal, can be represented simply by:

$$y = C \qquad \text{(II-78)}$$

or, if vertical, by:

$$x = g \qquad \text{(II-79)}$$

Substituting these, we immediately obtain:

$$r = -\frac{C - \frac{V_o + V}{2}}{\sin(\Theta)\cos\left(\tan^{-1}\left(\frac{V_o + V}{U_o + U}\right)\right) + \cos(\Theta)\sin\left(\tan^{-1}\left(\frac{V_o + V}{U_o + U}\right)\right)} \qquad \text{(II-80)}$$

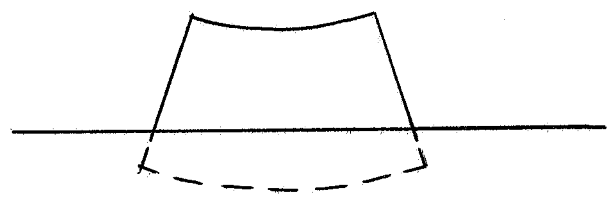
for a horizontal line and:

$$r = - \frac{g - \frac{U_o + U}{2}}{\cos(\Theta)\cos\left(\tan^{-1}\left(\frac{V_o + V}{U_o + U}\right)\right) - \sin(\Theta)\sin\left(\tan^{-1}\left(\frac{V_o + V}{U_o + U}\right)\right)} \quad \text{(II-81)}$$

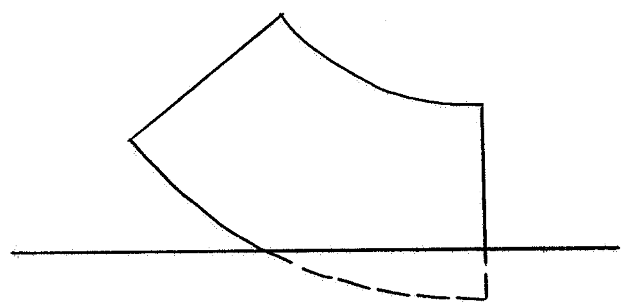
for a vertical line.

There are a number of possibilities or ways in which the horizontal line may intersect the mesh:

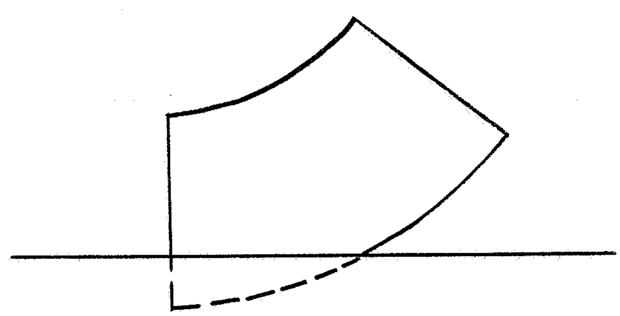
A)



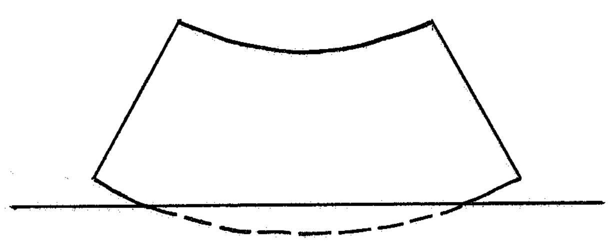
B)



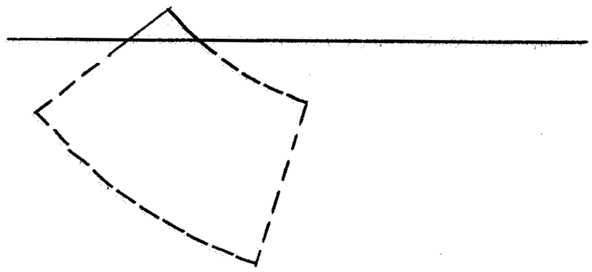
C)



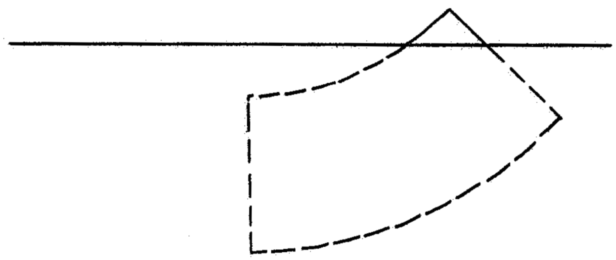
D)



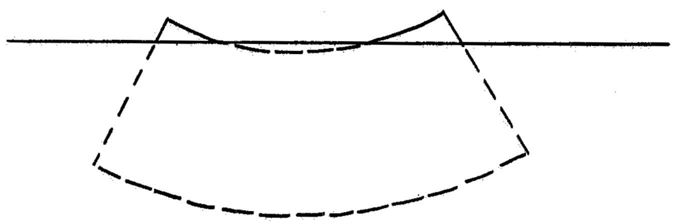
E)



F)



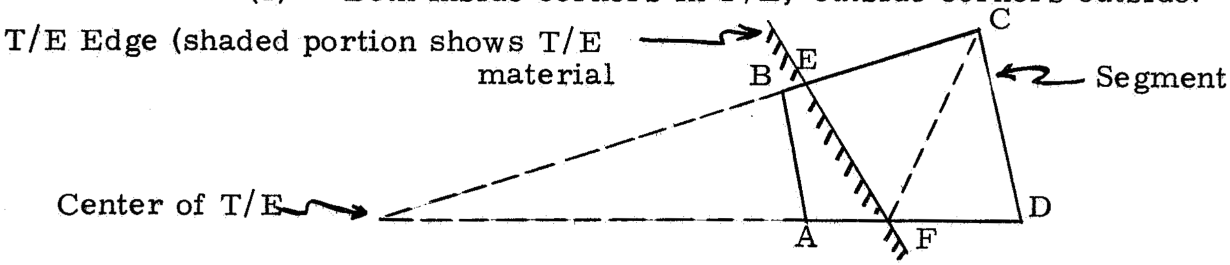
G)



In each case the area enclosed by the solid line must be computed and included within the mesh. Although these areas can be computed exactly, the size of the increments is small and small errors in the size of several increments will have negligible effect on the overall plate behavior. We will therefore assume the segment shape to be trapezoidal. This immediately eliminates cases D and G as having a negligible effect on the mesh.

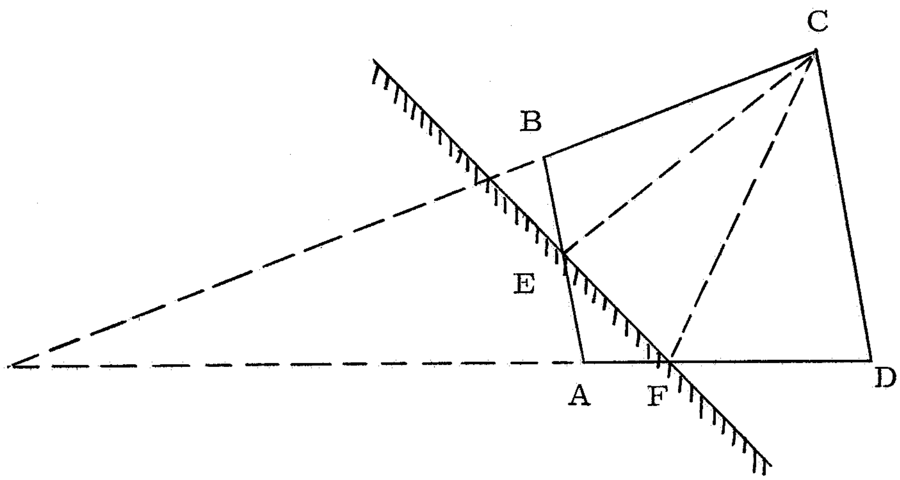
The following types of behavior can take place in fitting trapezoidal nodes to the plate-thermoelement intersection:

(1) Both inside corners in T/E, outside corners outside:



Solution: Redefine points A, B; treat as case 8.

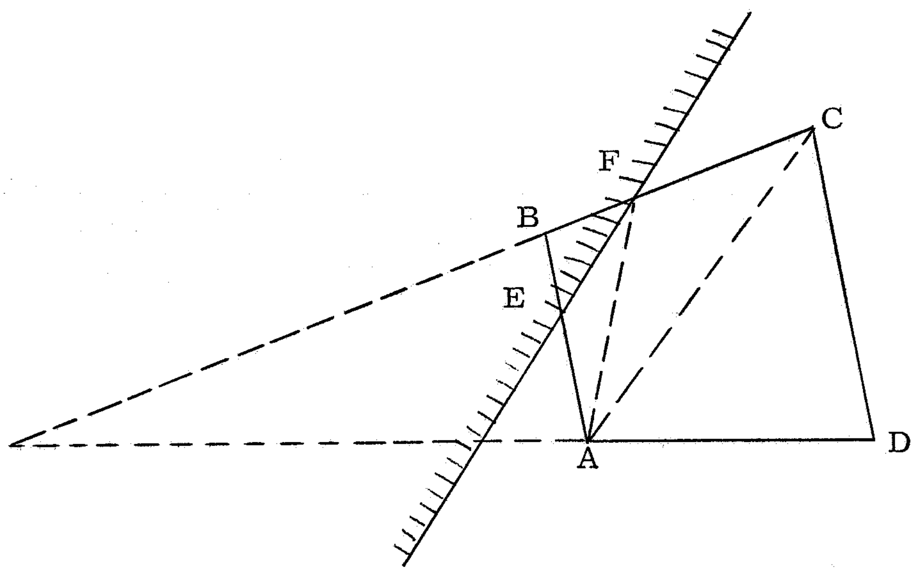
(2) Back inside corner in T/E, others outside:



Solution: Compute area for several triangles, and add to obtain the total:

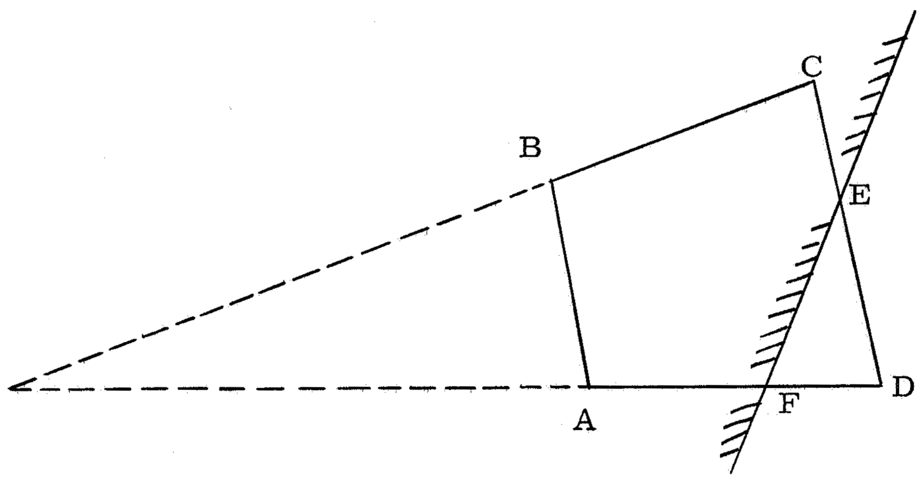
area = BCE + CEF + CDF (II-82)

(3) Forward inside corner in T/E, others outside:



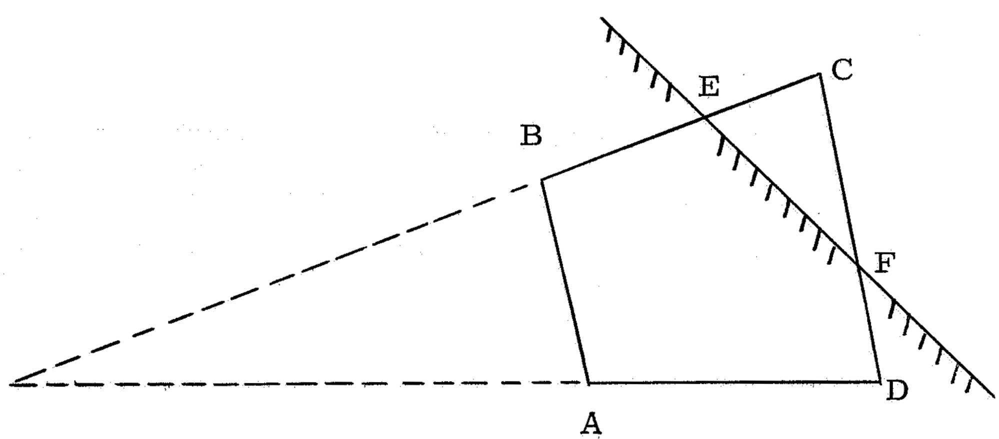
area = AEF + ACF + ACD (II-83)

(4) Back outside corner outside T/E, others inside:



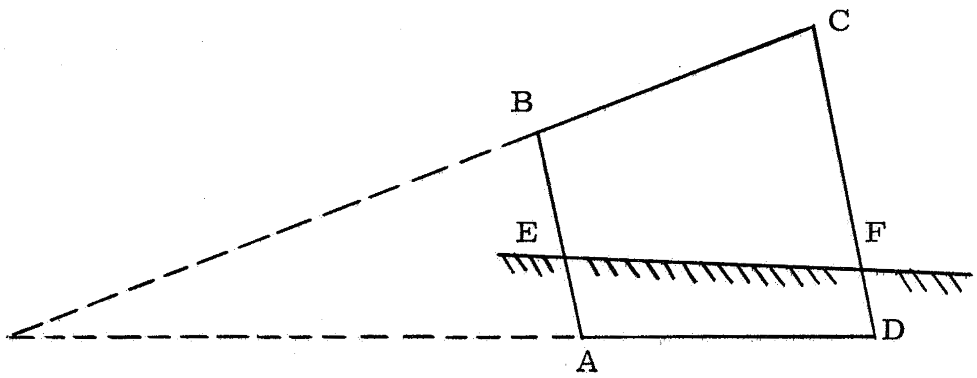
area = DEF (II-84)

(5) Forward outside corner outside T/E, others inside:



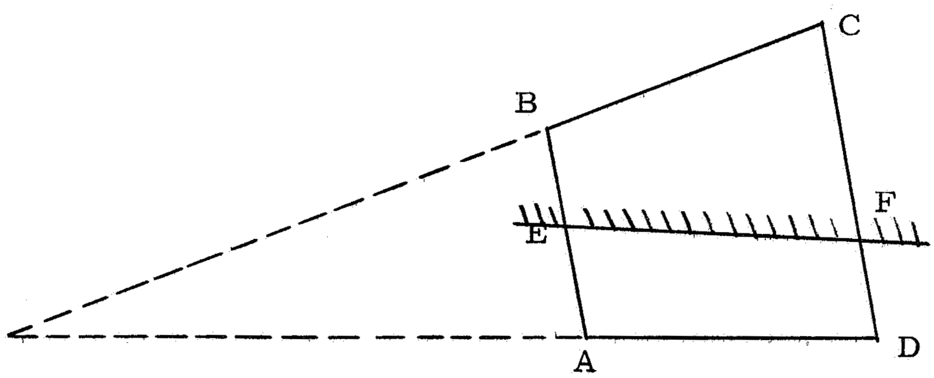
area = CEF (II-85)

(6) Backward corners in T/E, forward corners outside:



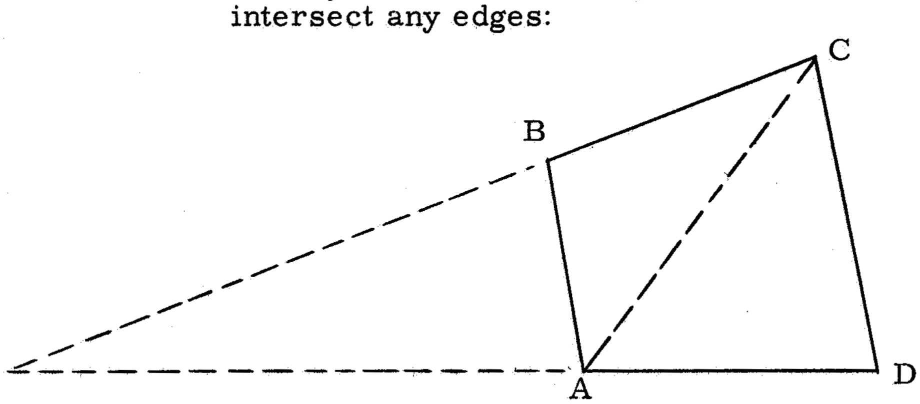
Redefine A, D; treat as case 8.

(7) Backward corners outside, forward corners in T/E:



Redefine B, C; treat as case 8.

(8) Finally, there is the simple case of a node that does not intersect any edges:



area = ABC + ACD (II-86)

Similar construction techniques can be used to describe node behavior along the plate edge. First refer to the same sketches and apply them to the plate edge. If we assume that the shaded side is within the plate, then the following may be defined:

Case	Area
1'	Redefine C, D, Treat as 8
2'	AEF
3'	BEF
4'	ABC + ACE + AEF
5'	ABE + AEF + ADF
6'	-
7'	-

There additionally exist the possibilities that a node may be common to the plate edge and the T/E edge, or that the plate edge and the T/E edge correspond. If they correspond, there is no node at that location. If the node is common to both edges, then there exist possibilities in the plate edge treatment where neither 1 - 7 or 1' - 7' are useful. It may be more convenient to subtract the position outside the plate edge from a previously computed area than to compute the area inside the plate directly. These cases will be treated using a double prime with the understanding that the indicated area is to be subtracted. The area in each case is described by the non-prime cases, but with the edge being the edge of the plate, not the edge of the T/E. In a few of these cases, it will also be convenient to merely move the points A B C D to correspond to the plate and T/E edges. In these cases, the treatment is the same as though no edges were present after the redefinition of the points has been made.

Since all nodes may be considered by considering triangles, we next consider the characteristics which must be calculated. The area of a triangle is:

$$A = \sqrt{s\,[s - a]\,[s - b]\,[s - c]} \tag{II-87}$$

where: $$s = \frac{1}{2}\,[a + b + c] \tag{II-88}$$

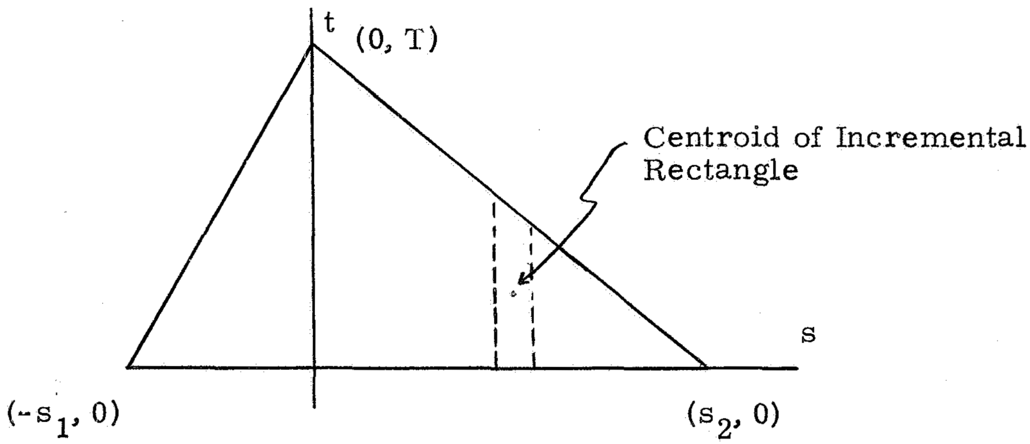
and a, b and c are the lengths of the sides. Once the coordinates of the corners have been obtained, the length of a line between the points may be determined from:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{II-89}$$

where d is the line length, and the line ends are at (x_2, y_2) and (x_1, y_1).

The selection of whether to treat the partial segment as a separate segment or as part of another segment may now be made. Some of these segments will be very small and if considered separately would require very small time steps. Consequently, we will make an arbitrary selection. If the area of a partial segment is less than one half that of the j-1 segment, then it will be attached to and made a part of the j-1 segment. This arbitrary selection is made with a small enough segment size that little error will be introduced by incorporating the segment into another segment.

The "center" of each of the segments modified by this procedure must be redetermined so that it is representative of the average behavior of the segment. We first determine the center of the partial segment without regard to its ultimate disposition. Since all areas were determined using triangles as a calculation base, we first determine the center of the triangle:



The moment of the incremental rectangle about the s axis is at $\frac{t}{2}$ dA = $\frac{t^2}{2}$ ds; about the t axis it is at s dA = stds. The equation of the left line is:

$$t = \frac{T}{S_1} s + T \tag{II-90}$$

and that of the right line:

$$t = +T - \frac{T}{S_2} s \tag{II-91}$$

The area is: $$A = \frac{T}{2}\left[S_1 + S_2\right] \tag{II-92}$$

The moment about the s axis is:

$$M_s = \int_{-S_1}^{0} \frac{t^2}{2} ds + \int_{0}^{S_2} \frac{t^2}{2} ds \tag{II-93}$$

Therefore:

$$M_s = \frac{1}{2}\int_{-S_1}^{0}\left[\frac{T}{S_1}s + T\right]^2 ds + \frac{1}{2}\int_{0}^{S_2}\left[T - \frac{T}{S_2}s\right]^2 ds \tag{II-94}$$

$$M_t = \int_{-S_1}^{0} stds + \int_{0}^{S_2} stds \tag{II-95}$$

$$M_t = \frac{T\left[S_2^{\ 2} - S_1^{\ 2}\right]}{6} \tag{II-96}$$

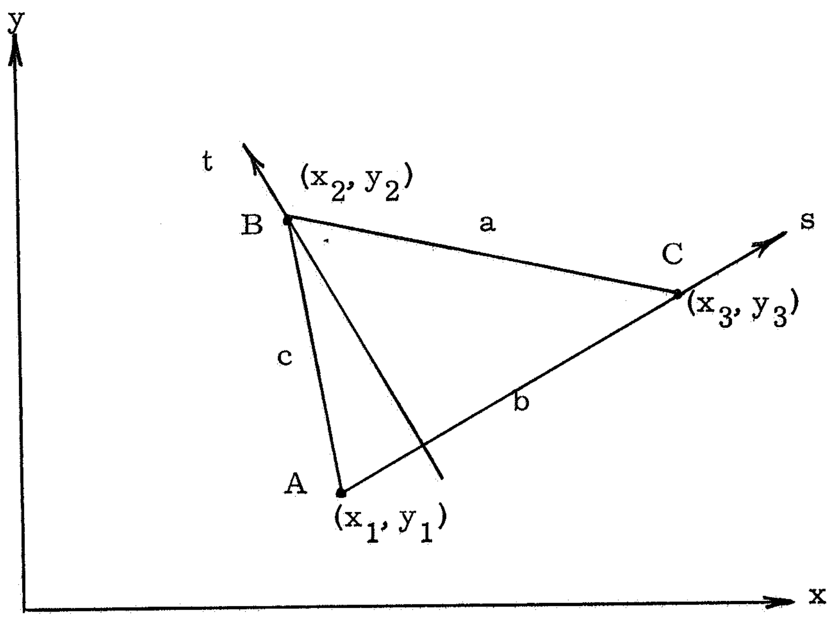
The centroids are now:

$$\overline{t} = \frac{2\ T^2\left[S_1 + S_2\right]}{6\ T\left[S_1 + S_2\right]} = \frac{T}{3} \tag{II-97}$$

$$\overline{s} = \frac{1}{3}\left[S_2 - S_1\right] \tag{II-98}$$

These centroids are based upon a triangle with an orientation as shown in the previous sketch. The orientations necessary here will be located on a different coordinate system with a different zero point and in most cases with a different rotational orientation. These results will not be applied using the (s, t) system, but the results are simple and can be easily applied to the coordinate system under consideration.

Now suppose the corners of the triangle are located at (x_1, y_1), (x_2, y_2) and (x_3, y_3). We will arbitrarily choose x_2 as corresponding to $s = 0$ and y_2 as corresponding to $t = T$. Further, (x_1, y_1) will correspond to $(-S_1, 0)$ and (x_3, y_3) to $(S_2, 0)$.



from the law of cosines:

$$\cos A = \frac{b^2 + c^2 - a^2}{2bc} \tag{II-99}$$

Further:

$$a = \sqrt{(x_2 - x_3)^2 + (y_2 - y_3)^2} \tag{II-100}$$

$$b = \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} \tag{II-101}$$

$$c = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{II-102}$$

$$S_1 = c \cos A \tag{II-103}$$

$$S_2 = b - S_1 \tag{II-104}$$

$$T = c \sin A \tag{II-105}$$

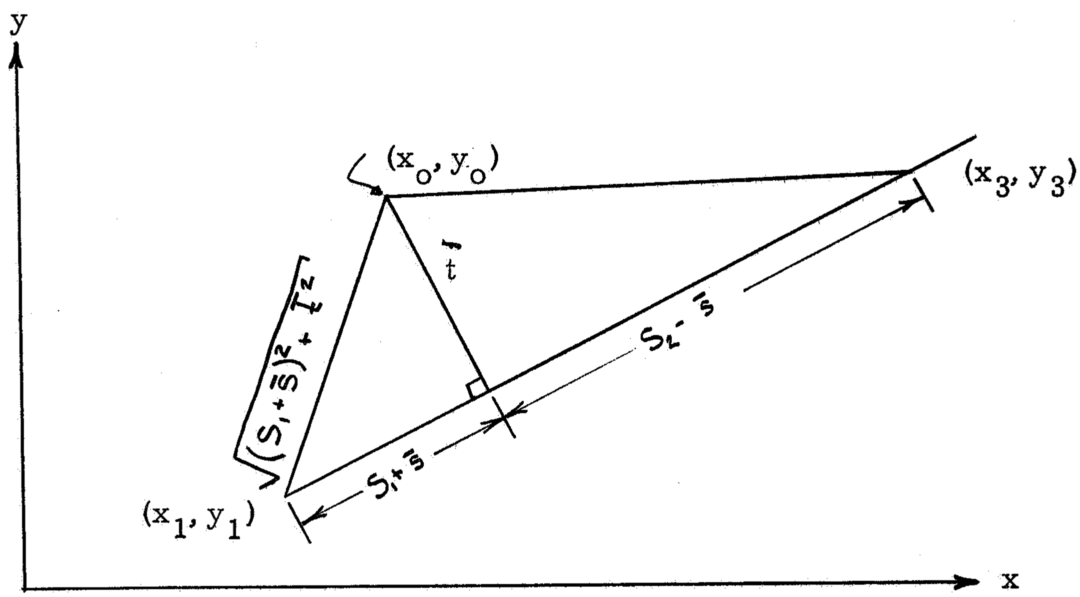
But:

$$\sin^2 A + \cos^2 A = 1 \tag{II-106}$$

$$\sin A = \sqrt{1 - \cos^2 A} \tag{II-107}$$

$$T = c\sqrt{1 - \cos^2 A} \tag{II-108}$$

With Equations (II-100) through (II-108) we can immediately find $\bar{s}$ and $\bar{t}$ from Equations (II-98) and (II-97). We may now convert these to the (x, y) coordinate system.



$$\sqrt{(S_1 + \bar{s})^2 + \bar{t}^{\,2}} = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2} = \sqrt{D} \tag{II-109}$$

$$\sqrt{(S_2 - \bar{s})^2 + \bar{t}^{\,2}} = \sqrt{(x_3 - x_0)^2 + (y_3 - y_0)^2} = \sqrt{G} \tag{II-110}$$

$$x_1^{\,2} - 2x_0\, x_1 + x_0^{\,2} + y_1^{\,2} - 2y_0\, y_1 + y_0^{\,2} = D \tag{II-111}$$

$$x_3^{\,2} - 2x_0\, x_3 + x_0^{\,2} + y_3^{\,2} - 2y_0\, y_3 + y_0^{\,2} = G \tag{II-112}$$

$$x_0 = \frac{D - G - x_1^{\,2} + x_3^{\,2} - y_1^{\,2} + y_3^{\,2} + 2y_0 \left[y_1 - y_3\right]}{-2\left[x_1 - x_3\right]} \tag{II-113}$$

This can be written as:

$$x_0 = E + F\, y_0 \tag{II-114}$$

and we may substitute this into Equation (II-111) to obtain:

$$x_1^2 - 2x_1 \left[E + Fy_0\right] + \left[E + Fy_0\right]^2 + y_1^2 - 2y_0\, y_1 + y_0^2 = D \tag{II-115}$$

$$\left[F^2 + 1\right] y_0^2 + \left[-2x_1 F + 2E\, F - 2y_1\right] y_0 +$$

$$+ x_1^2 - 2x_1 E + E^2 + y_1^2 - D = 0 \tag{II-116}$$

which can be written:

$$a\, y_0^2 + b\, y_0 + C = 0 \tag{II-117}$$

with the solution:

$$y_0 = \frac{-\,b \pm \sqrt{b^2 - 4\,ac}}{2a} \tag{II-118}$$

To determine which solution is to be used we first note that y_0 must be between y_1 and y_3, y_1 and y_2, or y_2 and y_3; and x_0 must be between x_1 and x_3, x_1 and x_2, or x_2 and x_3. It is not likely both the y_0's will satisfy this requirement and we will not investigate this further unless a problem is found to exist in the actual calculations.

There can be a problem if $x_1 = x_3$ because of a zero in the denominator of Equation (II-113). In this case we write:

$$y_1^2 - y_3^2 - 2\,y_0 \left[y_1 - y_3\right] = D - G \tag{II-119}$$

$$\frac{y_1^2 - y_3^2 - D + G}{2\quad y_1 - y_3} = y_0 \tag{II-120}$$

$$x_0^2 + (-2x_1)\, x_0 + (x_1^2 + y_1^2 - 2y_0\, y_1 + y_0^2 - D) = 0 \tag{II-121}$$

and:

$$x_0 = \frac{-\,bt \pm \sqrt{bt^2 - 4 * at * ct}}{2. * at} \qquad \text{(II-122)}$$

With the "center" of the triangle determined to be (x_0, y_0), we may now determine the new center of the segment. This will be located at:

$$x_s = x_n A_n + x_0 A_t \qquad \text{(II-123)}$$

$$y_s = y_n A_n + y_0 A_t \qquad \text{(II-124)}$$

where: s refers to the edge segment center

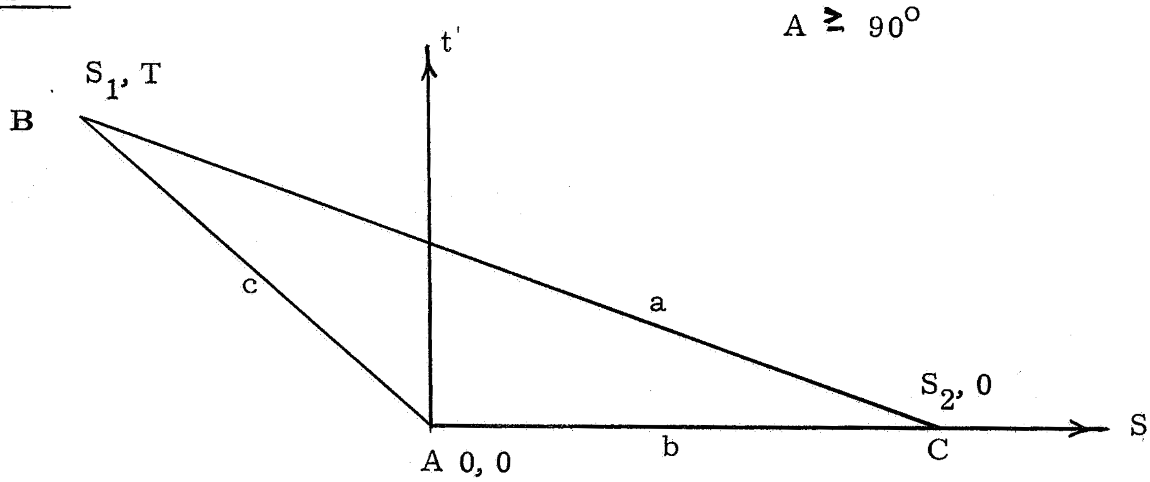
n refers to the original node

A_t = triangle area, positive if the triangle is to be added, negative if removed from the original node.

If a trapezoid is to be added or removed, the location of the center is a simple addition of two triangles and will not be discussed here.

Two other "types" of triangles can result during the calculation process, which will be referred to as Cases 2 and 3. Similar investigations result in the following:

Case 2



$$s_1 = c\,(-\cos A) \qquad \text{(II-125)}$$

$$s_2 = b \qquad \text{(II-126)}$$

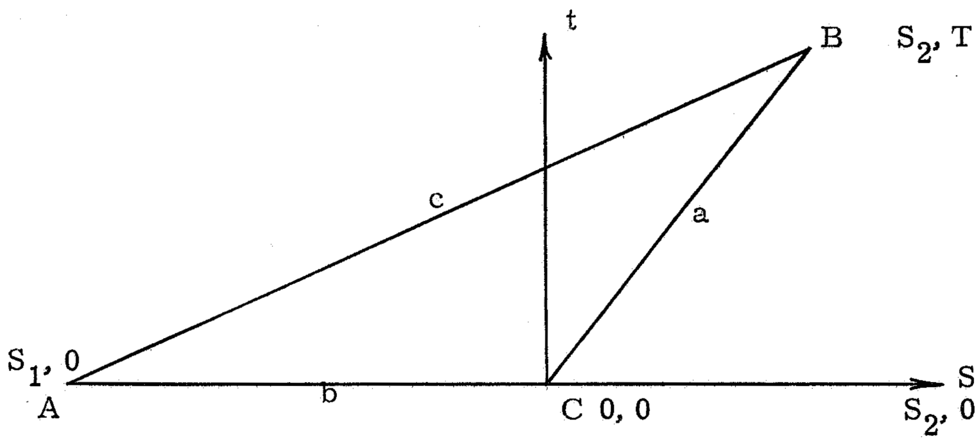
$$T = c\sqrt{1 - \cos^2 A} \qquad \text{(II-127)}$$

$$\overline{s} = \frac{s_1\, s_2 - s_1^{\,2}}{3\,(s_1 + s_2)} \tag{II-128}$$

$$\overline{t} = \frac{T\, s_1}{3\,(s_1 + s_2)} \tag{II-129}$$

Case 3

$C \geq 90^o$

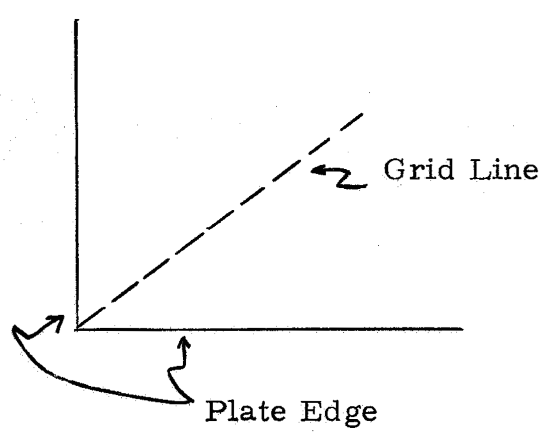


$$s_1 = b \tag{II-130}$$

$$s_2 = a\,(-\cos C) \tag{II-131}$$

$$T = a \sqrt{1 - \cos^2 C} \tag{II-132}$$

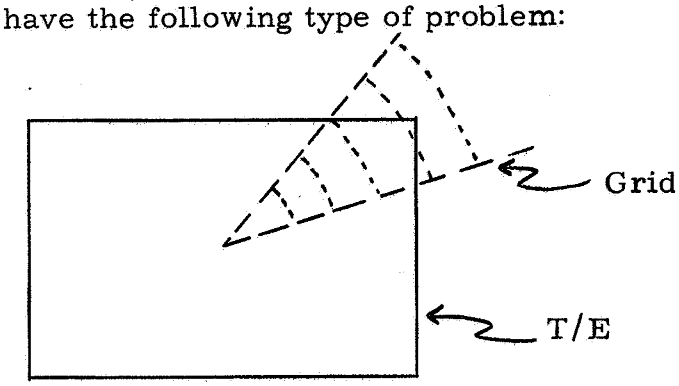
Plate corners will introduce no problem in mesh construction because of the angular construction technique. In all cases along the outside edges, the node construction is of the type:



$$\overline{S} = \frac{S_2^2 - S_2 S_1}{3(S_2 + S_1)} \tag{II-133}$$

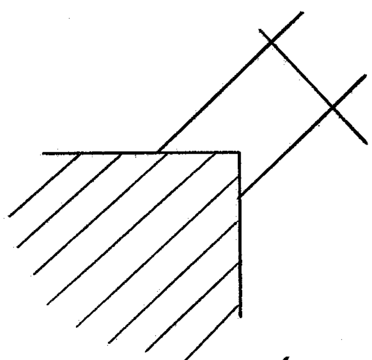
$$\overline{t} = \frac{t}{3}\ \frac{S_2}{S_1 + S_2} \tag{II-134}$$

for all four corners. However, this is not the case for the thermoelement edge, and here we can have the following type of problem:

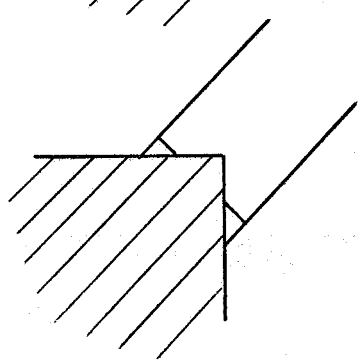


which must be considered. The following cases can result:

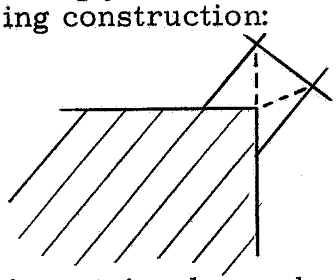
(1)



(2)



Case two will be treated by merely adding the triangular segments to the adjoining segment (the increased r position) and adjusting the segment center location accordingly (as previously discussed). Case one can be considered by the following construction:



which gives three triangles and which may be immediately considered by the techniques already discussed.

Unfortunately, the ability to compute the various node characteristics is not worthwhile if the computer cannot be instructed to select the type of node.

To determine the type of node under consideration, we first compare point A on the trapezoid to the thermoelement edge at the same angular location and the corresponding plate edge. An index value, n_A, is assigned according to the following:

Location	Index, n_A
In plate	1
In T/E	2
Outside plate	3

Values on an edge are treated as though they were in the plate. Similar assignments are then made for points B, C and D. We may then compute a "case index", MZU, according to the equation:

$$MZU = n_D + 3\,(n_C - 1) + 9\,(n_B - 1) + 27\,(n_A - 1) \qquad \text{(II-135)}$$

There results the values shown in Table II-2. The item is the case to be treated, and these are defined in Table II-3. The item numbers also correspond to the source program statement numbers in subroutine MESHA. The branching is performed by using MZU as the index for a computed go to statement (see source statement number 24).

TABLE II-2. CORNER TREATMENT CHARACTERISTICS

n_A	n_B	n_C	n_D	MZU	Item	n_A	n_B	n_C	n_D	MZU	Item
1	1	1	1	1	36	2	2	2	3	42	56
1	1	1	2	2	37	2	2	3	1	43	57
1	1	1	3	3	38	2	2	3	2	44	58
1	1	2	1	4	37	2	2	3	3	45	59
1	1	2	2	5	37	2	3	1	1	46	37
1	1	2	3	6	37	2	3	1	2	47	37
1	1	3	1	7	39	2	3	1	3	48	37
1	1	3	2	8	37	2	3	2	1	49	37
1	1	3	3	9	40	2	3	2	2	50	37
1	2	1	1	10	41	2	3	2	3	51	37
1	2	1	2	11	37	2	3	3	1	52	70
1	2	1	3	12	42	2	3	3	2	53	60
1	2	2	1	13	68	2	3	3	3	54	61
1	2	2	2	14	37	3	1	1	1	55	37
1	2	2	3	15	69	3	1	1	2	56	37
1	2	3	1	16	37	3	1	1	3	57	62
1	2	3	2	17	37	3	1	2	1	58	37
1	2	3	3	18	43	3	1	2	2	59	37
1	3	1	1	19	37	3	1	2	3	60	37
1	3	1	2	20	37	3	1	3	1	61	37
1	3	1	3	21	37	3	1	3	2	62	37
1	3	2	1	22	37	3	1	3	3	63	63
1	3	2	2	23	37	3	2	1	1	64	37
1	3	2	3	24	37	3	2	1	2	65	37
1	3	3	1	25	44	3	2	1	3	66	64
1	3	3	2	26	37	3	2	2	1	67	37
1	3	3	3	27	45	3	2	2	2	68	37
2	1	1	1	28	46	3	2	2	3	69	65
2	1	1	2	29	47	3	2	3	1	70	37
2	1	1	3	30	37	3	2	3	2	71	37
2	1	2	1	31	37	3	2	3	3	72	66
2	1	2	2	32	37	3	3	1	1	73	37
2	1	2	3	33	37	3	3	1	2	74	37
2	1	3	1	34	48	3	3	1	3	75	37
2	1	3	2	35	49	3	3	2	1	76	37
2	1	3	3	36	50	3	3	2	2	77	37
2	2	1	1	37	51	3	3	2	3	78	37
2	2	1	2	38	52	3	3	3	1	79	37
2	2	1	3	39	53	3	3	3	2	80	37
2	2	2	1	40	54	3	3	3	3	81	67
2	2	2	2	41	55						

TABLE II-3. ITEM DEFINITIONS

Item (MESHA Source Program Statement Number)	Case
36	8
37	impossible mesh specification
38	4'
39	5'
40	1'
41	3
42	3 - 4''
43	3 - 1''
44	6'
45	2'
46	2
47	6
48	2 - 5''
49	6 - 5''
50	2 - 1''
51	1
52	5
53	1 - 4''
54	4
55	all T/E
56	4 - 4''
57	1 - 5''
58	5 - 5''
59	1 - 1''
60	6 - 6''
61	2 - 2''
62	7'
63	3'
64	3 - 7''
65	7 - 7''
66	3 - 3''
67	all outside
68	7
69	7 - 4''
70	2 - 6''

D. Steady State Calculations

Equation (II-16) may be rewritten in the following form in the steady state:

$$\frac{k_{n+1} + k_n}{2 \Delta Z} \left[T_{n+1} - T_n \right] - \frac{k_n + k_{n-1}}{2 \Delta Z} \left[T_n - T_{n-1} \right] + \\ + \frac{\rho_n I^2 \Delta Z}{A^2} - \frac{T_n I}{2A} \left[\alpha_{n+1} - \alpha_{n-1} \right] - \frac{F P}{A} \\ \left[T_n^4 - \overline{T_s^4} \right] = 0 \qquad \text{(II-135)}$$

This can be rewritten as:

$$\frac{k_{n+1} + k_n}{2} \left[T_{n+1} - T_n \right] - \frac{k_n + k_{n-1}}{2} \left[T_n - T_{n-1} \right] + \frac{\rho_n I^2 \Delta Z^2}{A^2} - \\ - \frac{T_n I \Delta Z}{2A} \left[\alpha_{n+1} - \alpha_{n-1} \right] - \frac{F P}{A} \left[T_n^4 - \overline{T_s^4} \right] = 0 \qquad \text{(II-136)}$$

or:

$$T_n = \left[\frac{\rho_n I^2 \Delta Z}{A^2} - \frac{T_n I \Delta Z}{2A} \left(\alpha_{n+1} - \alpha_{n-1} \right) - \frac{F P}{A} \right. \\ \left. \left(T_n^4 - \overline{T_s^4} \right) + \frac{(k_n + k_{n+1})}{2} T_{n+1} + \frac{(k_n + k_{n-1}) T_{n-1}}{2} \right] \Bigg[\\ \left. \frac{k_n + k_{n+1}}{2} + \frac{k_n + k_{n-1}}{2} \right]^{-1} \qquad \text{(II-137)}$$

Equation (II-137) forms the basis for the steady state calculation for the thermoelement.

The absorber and the radiator may be considered by rewriting Equation (II-36) for the steady state situation:

$$\frac{Q_{i,j}}{t_p} - \frac{\sigma F_{i,j}}{t_p r_j \Delta \Theta \Delta r} \left[T_{i,j}^4 - \overline{T'^4} \right] + S_{i,j} - \sigma \epsilon_{o_{i,j}} T_{i,j}^4 +$$

$$+ \frac{1}{r_j \Delta\Theta \Delta r^2} \left\{ \Delta\Theta k_{i,j} \left[r_j - \frac{\Delta r}{2} \right] \left[T_{i,j-1} - T_{i,j} \right] + \right.$$

$$+ \Delta\Theta k_{i,j+1} \left[r_j + \frac{\Delta r}{2} \right] \left[T_{i,j+1} - T_{i,j} \right] +$$

$$+ k_{i,j} \frac{\Delta r^2}{r_j \Delta\Theta} \left[T_{i+1,j} - T_{i,j} \right] + \frac{k_{i,j} \Delta r^2}{r_j \Delta\Theta}$$

$$\left. \left[T_{i-1,j} - T_{i,j} \right] \right\} = 0 \qquad \text{(II-138)}$$

where the thermal conductivity behavior has been approximated. Now consider the following:

$$k_{i,j} \frac{\left[r_j - \frac{\Delta r}{2} \right] \Delta\Theta t_p}{\Delta r} = Y_{r_{i,j}} = \text{radial conductance} \qquad \text{(II-139)}$$

$$k_{i,j+1} \frac{\left[r_j + \frac{\Delta r}{2} \right] \Delta\Theta t_p}{\Delta r} = Y_{r_{i,j+1}} = \text{radial conductance} \qquad \text{(II-140)}$$

$$k_{i,j} \frac{\Delta r t_p}{r_j \Delta\Theta} = Y_{\Theta i,j} = \text{angular conductance} \qquad \text{(II-141)}$$

$$r_j \Delta\Theta \Delta r = A_{i,j} = \text{surface area} \qquad \text{(II-142)}$$

Use of these equations allows Equation (II-138) to be rewritten as:

$$Q_{i,j} - \frac{\sigma F_{i,j}}{A_{i,j}} \left[T_{i,j}^4 - \overline{T'}^4 \right] + S_{i,j} t_p - \sigma \epsilon_{o\,i,j} t_p T_{i,j}^4 +$$

$$+ \frac{1}{A_{i,j}} \left\{ Y_{r\,i,j} \left[T_{i,j-1} - T_{i,j} \right] + Y_{r_{i,j+1}} (T_{i,j+1} - T_{i,j}) + \right.$$

$$+ Y_{\Theta_{i+1,j}} \left[T_{i+1,j} - T_{i,j} \right] + Y_{\Theta_{i,j}} \left[T_{i-1,j} - T_{i,j} \right] \Bigg\} = 0 \qquad \text{(II-143)}$$

or:

$$T_{i,j} = A_{i,j} \left[Q_{i,j} - \frac{\sigma F_{i,j}}{A_{i,j}} \left[T_{i,j}^{4} - \overline{T'}^{4} \right] + S_{i,j}\, t_p - \right.$$

$$- \sigma \epsilon_{o\, i,j}\, t_p\, T_{i,j}^{4} + \frac{1}{A_{i,j}} \Bigg\{ Y_{r_{i,j}} T_{i,j-1} + Y_{r_{i,j+1}} T_{i,j+1} +$$

$$\left. + Y_{\Theta_{i+1,j}} T_{i+1,j} + Y_{\Theta_{i,j}} T_{i-1,j} \Bigg\} \right] \Bigg[Y_{r_{i,j}} + Y_{r_{i,j+1}} +$$

$$+ \left. Y_{\Theta_{i+1,j}} + Y_{\Theta_{i,j}} \right]^{-1} \qquad \text{(II-144)}$$

Equation (II-144) represents the form utilized for the steady state calculations of the absorber or the radiator temperatures. Although this equation has been derived by considering the difference equation which describes the plate behavior, it is virtually identical to the one obtained by the computer program from the automatically generated mesh arrangement. The only basic differences pertain to the determination of the conductance, which in the computer program, is obtained with a variable increment size in the radial and angular directions.

Investigation of Equations (II-137) and (II-144) show that the temperature which appears on the left side of the equation also appears on the right side to the fourth power. This is an unstable configuration if the term contributes to a significant degree to the solution. Normally, the contribution in Equation (II-137) is small and the effect is damped so that instability does not occur except in extreme circumstances. This is not the case when Equation (II-144) is applied to the radiator, although a problem probably will not occur in the absorber. The radiator rejects virtually all of the heat associated with the system to the surroundings via this temperature term to the fourth power. This represents an extremely unstable form. The instability will normally take the appearance of calculating a temperature on the left side which is too low, followed by the recalculation using the new value on the right side which gives a temperature on the left that is too high, followed again by one which is too low, with divergence occurring very rapidly. Damping of this effect with this alternating type behavior is relatively easy to obtain by incorporating a fractional portion of the new value as computed by Equation (II-144) into the value which appears on the right side. This approach is utilized in the computer program as convergence to the correct answers is obtained.

It is not desirable to use strong damping during the entire calculation because this slows convergence to the correct answer. However, criteria still must be applied with this type of an equation so that stability is assured if it is not damped. The calculation approach utilized herein was to assume an allowed temperature change per iteration, and damp or extrapolate the results accordingly. The application of this process to the solutions may be obtained by considering:

$$T_{new} = T_{old} + f\,(T_{old}) + C\,T_{old}^{\;4} \qquad \text{(II-145)}$$

where:

T_{new} = new value of temperature

T_{old} = previous value of temperature

f = a linear function of T_{old} which need not be defined here

C = a constant

which is the type of equation that must be handled. Since the temperature change per iteration is:

$$\Delta T = T_{new} - T_{old} \qquad \text{(II-146)}$$

Equation (II-145) becomes:

$$\Delta T = f\,(T_{old}) + C\,T_{old}^{\;4} \qquad \text{(II-147)}$$

Now consider that the calculation is to be forced so that a change of $\Delta T'$ is to occur per iteration, and that a number of equations of the type of (II-147) are applied to different nodes with some other node temperatures appearing on the right side. Then each of the ΔT_i (for the ith node) may be computed. If ΔT_{max} is the largest of the ΔT_i, then all of the ΔT_i are recalculated according to:

$$\Delta T_{i_{new}} = \Delta T_{i_{old}} \frac{\Delta T'}{\Delta T_{max}} \qquad \text{(II-148)}$$

Immediately, the new temperature for the next iteration may be obtained from:

$$T_{new_i} = T_{old_i} + \Delta T_{i_{new}} \qquad \text{(II-149)}$$

This type of iteration will never converge to steady state answers because it automatically extrapolates so that the greatest temperature change is equal to $\Delta T'$. Consequently, as the correct value is approached the forcing technique represented by Equations (II-148) and (II-149) is removed from the calculation procedure and a damping factor, as previously discussed, is applied. The point at which this occurs in the computer program is determined from the user's estimate of how many iterations will be required to reach steady state. This means that for most problems the user's estimate of the number of iterations will represent the minimum number of iterations the code will perform. It will virtually always require more iterations than estimated before it will automatically terminate the calculation procedure. If the termination occurs shortly after the estimated number of iterations has been reached, then the estimated value for future problems of that type should be reduced.

III. COMPUTER PROGRAM USE INSTRUCTIONS

A. Data Input

1. Introduction

Input for this program is flexible and for practical purposes need not follow any particular format. The location of the data on the cards, the number of items per card, the field length, etc., are entirely at the discretion of the user. Columns 1-72 may be used for input information. Other restrictions are that individual numbers must be separated by at least one blank, and one number may not be continued from one card to another. Extra blanks can be included as desired, since they will be ignored by the code. In general, however, the more compact the input the faster will be the data processing.

There is no designation between fixed point and floating point variables in the input. Hence, the following may all be used to read a "one" into the machine:

```
1
1.
1.0
1.0E+0
10.E-1
1.+0
10.-1
1.000000
+1
+10.-1
.1E+1
0.10E+1
+0.1E+1
0.1+1
etc.
```

The only restriction for inputting numbers, aside from number separation by blanks, is that numbers greater than approximately 32,000 must be specified in an exponential form.

This computer program is written in such a manner that it may be readily expanded. Therefore, the input sections of the program have been programmed so that more information can be provided than is presently utilized. The input description is written to be compatible with the existing input routines, and where information that is provided in the input is not utilized, comments are made to that effect. This approach has been used so that as additions are made to the program, the use instructions may be kept up to date by merely deleting the necessary comments.

The input data are divided into major groups, which are in turn divided into small groups or individual items. Specification of the group (for example *3 for group 3) and a subgroup number defines the data to follow. Property tables may be read in any order, and the length of the table need not be given. The code will recognize the end of tables from the input. Groups and subgroups may be read in any desired order. When the end of input data is signaled, the code will perform a logical check to be sure that all of the necessary data have been read and that no conflicts have been specified. The data tables will be placed in the proper order and calculations begun.

Following problems may be investigated very easily when parametric investigations are to be performed. In these cases it is only necessary to read those data that are to be changed.

The panel may be defined by reading the overall dimensions of each of the constituents in terms of an (x, y) coordinate system oriented at one corner of the panel, and by reading thermoelement length. Specification of the number of divisions desired for computational purposes along each major dimension is sufficient for setting up a mesh. The code utilizes this information to generate a mesh approximately conforming to the input specifications.

2. Control Cards

Control cards are designated with a * in column one and a number in column two. The remainder of the card may be used for input information if desired. The following designations have been established:

Control Card	Meaning
*1	Descriptive Information
*2	Dimensions
*3	Thermoelement Properties
*4	Other Properties
*5	Heating Information
*6	Specifications
*7	Input complete - Partial input for another problem follows
*8	Input complete, complete input for another problem follows
*9	Input complete, no input follows

With the exception of *7 - *9 cards, the information may be read in any desired order. The *7 - *9 cards are used to signal the end of input for a problem and are therefore located last. The *1 - *6 cards may be read in any order and, if desired, partial information in one group may be read. This makes possible such input options as reading some *2 information, then some *5 information, then some more *2, etc. Further, if only one item is to be changed between problems, this may be done by reading only that one item. Other information previously read into the machine will be saved unchanged with the exception of *1 information.

It is never necessary to specify how much information is to be read before reading. Termination of a group of information will occur if a "Z" is inserted following the data. This can appear anywhere on a card, and can be followed on the same card by other numerical information. The first number following the Z must be a control number (as discussed under the * headings). A Z may not be used if the information to be read consists of a fixed number of words.

3. *1, Descriptive Information

Information read by the *1 card is printed as output at the beginning of the problem for which it is read and is then destroyed. The first card of this group must contain *1 in columns 1 and 2. Columns 3-72 may be used for any title information that is desired. Following cards do not need *1 in columns 1 and 2 although no harm will be done if this is included. In either case, only columns 3-72 will be printed as output. As many cards as desired may be read and printed in this manner.

It is not necessary to reread data already in the machine since this information is utilized automatically. (See *7 and *8 for further discussion of this point).

4. *2, Dimensions

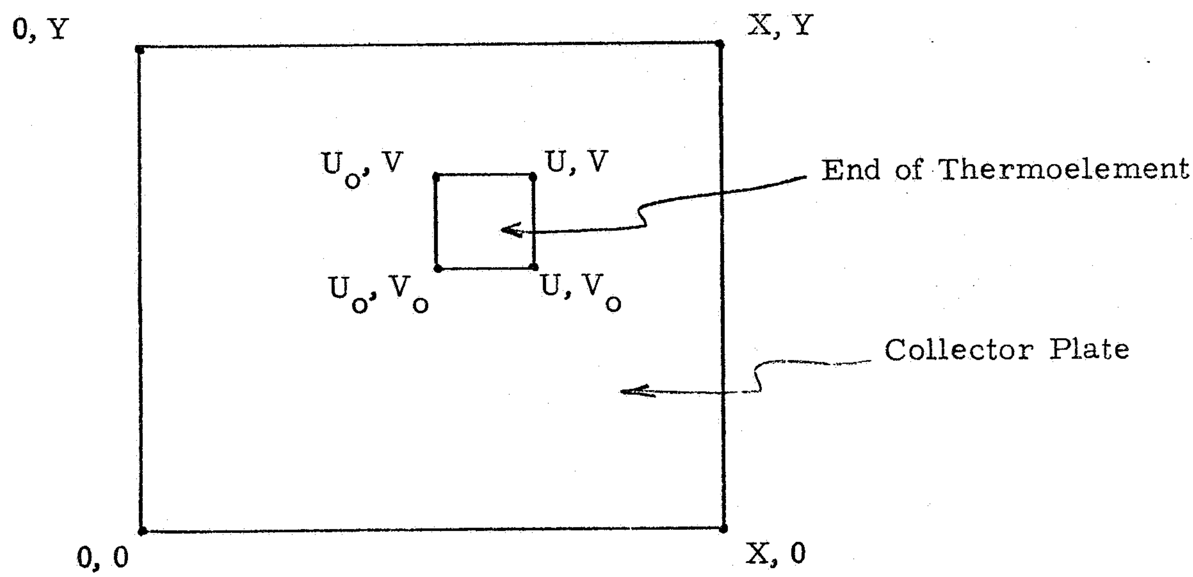
The first number read following *2 is a control number, which will be referred to as C throughout the remainder of this report. The value for C is then followed by the required information as specified in this section and summarized in Table 1. Another C may follow the presented information or a new control card may be used. The C's may be read in any desired order and as few C's as desired may be used, subject to the restriction that the dimensions must be fully described or the information must have been previously read.

TABLE III-1.* *2 CONTROL NUMBERS (C)

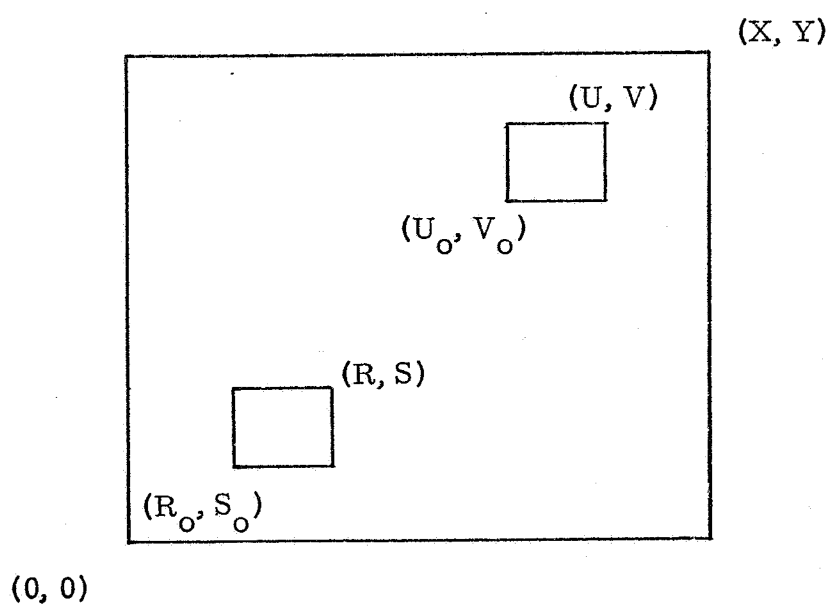
C Value	Data Required
1	Collector thickness
2	Collector dimensions X, Y and Z
3	Thermoelement dimensions Uo, Vo, U and V
4	Thermoelement dimensions Uo, Vo, U, V, Ro, So, R, S
5	Thermoelement length for non-segmented thermoelements
6	Number of segments followed by lengths for segmented elements
7	Thermoelement material designations
8	Radiator and edge dimensions
9	Hot shoe thickness and material designations
10	Cold shoe thicknesses and material designations
11	Mesh information
12	Boundary information

* The type 2 information used by the program in the present version consists of X and Y. Z must be read, but is not used for calculations. The options offered under the type 4 and type 6 input are not available, and C values corresponding to these should not be used. The type 8 information pertains to radiator edge thickness and edge width as well as radiator thickness. Only the radiator thickness is used in the calculations, although all 3 values must be provided for proper functioning of the input. The type 9 and type 10 information is used only to provide contact resistance data in the present version.

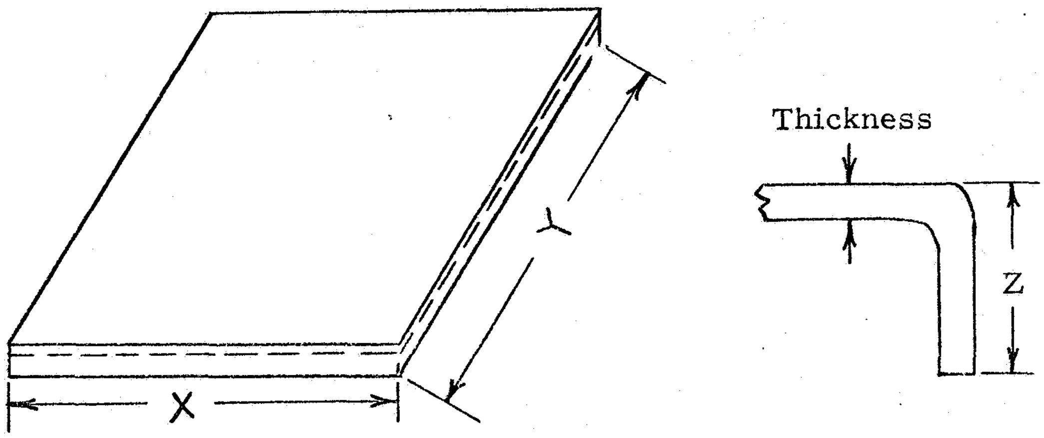
Two types of collector plates can be considered:



and a similar arrangement with two attached thermoelements rather than one. In the latter case the thermoelement corners will be specified by R and S as well as U and V. The meaing of R is the same as U, that of S the same as V, except we now describe two thermoelements:



Only one piece of information will be read with a C=1 type input, the absorber (or hot side fin) thickness. Any dimensions may be used as long as a consistent set of data are given for all of the input. Type C=2 information reads the collector dimensions X, Y and Z (in that order) as shown on the following sketch:



Note that Z is not the thickness, but the total outside dimension of the plate edge that has been bent over.

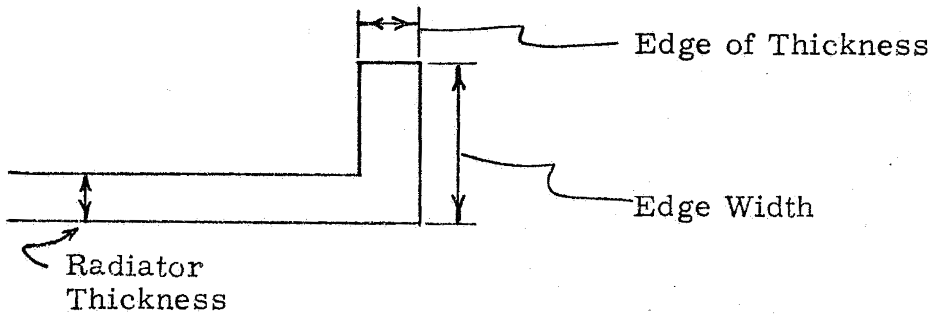
Type C=3 and 4 information refers to the thermoelement dimensions as previously discussed. This information will be read in the order presented under the headings in Table III-1. If for some reason both 3 and 4 type information were read, an error would result and this case would be skipped. Further, if type 3 information had been read for a previous problem, and type 4 information read for a present problem, then sufficient additional information would be required that the problem were fully described or the case would be refused. This will be discussed further at the end of this subsection.

Type C=5 information requires the thermoelement length for non-segmented thermoelements. If C=3 information has been (or is to be) read, one length is required; if C=4, two lengths, the first for the UV T/E, the second the RS T/E. The end of this information must be signaled by a "Z" since the code "doesn't know" how many numbers to read.

Type C=6 information is used if segmented thermoelements are to be used. The first number read is the number of segments in the UV system (10 or less are allowed) and this is followed by the segment lengths. If information for an RS system is required, this is read next in the same manner. The end of this information must be signaled by a "Z". The segments should be read beginning at the hot end.

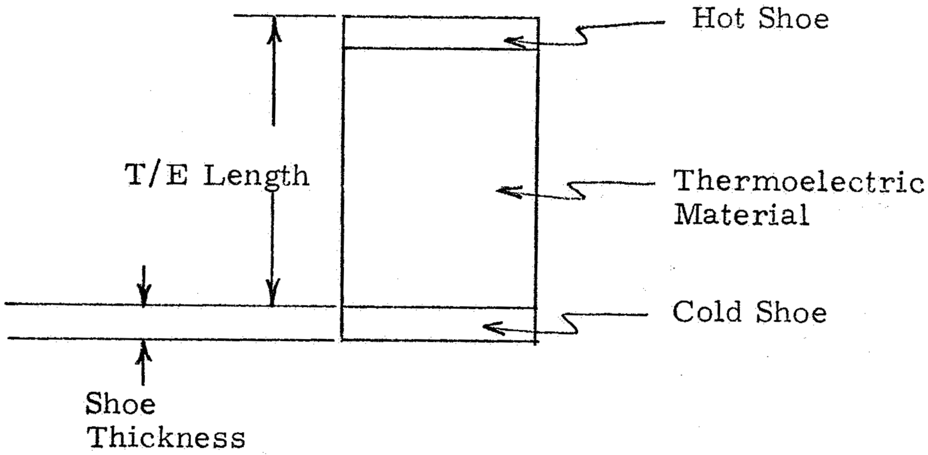
Type C=7 are thermoelement material designations read in the same order as for type 6. These material designations are numbers referring to properties read using *3 input and, therefore, specify the thermoelement material. A "Z" must terminate this information.

Type C=8 input is three numbers, the radiator thickness, the edge width, and the edge thickness.



These are read in the presented order.

Each thermoelement or thermoelement segment is considered to have the following form:



C=9 is used to read in the hot shoe thickness followed by the hot shoe material designation. If multiple thermoelements or segments are used, the readings are repeated in the order read in C=7. All thicknesses are read first, and then all material numbers. If R-S as well as U-V thermoelements are used, the thicknesses for the U-V T/E's are read, then for the R-S T/E's. The material designations are then read in the same order. C=10 does the same thing for the cold shoes. Note that zero thickness shoes are allowed, which is the same as no shoe. A material still must be specified, however. The end of C=9 and of C=10 information must be specified by using a "Z".

Mesh information is given through the use of C=11. The number of mesh points are read for each of the dimensions in the following order:

(1) Each collector side starting at (0,0) and proceeding in a counter-clockwise direction.

(2) Each collector side starting at (Uo, Vo) and proceeding as before.

(3) Repeat Item 2 (above) for R S if used, otherwise this is ignored.

(4) The number of mesh points in the thermoelements reading first the UV T/E and then the R S T/E. If segmented T/E's are used, begin at the hot end and work toward the cold end reading the number of mesh points per segment for the UV T/E. Repeat for the R/S T/E if one is used.

These data are used as guides in setting up the mesh for calculation purposes. They will seldom be used exactly as specified. The total number of mesh points must be less than 1000 (the collector and radiator have the same number of points). The end of this information must be indicated by a "Z".

The C=12 information pertains to whether the edges are exposed or "see" other flat plate packages and hence are insulated. Ones should be read if the edges are exposed, zeroes if insulated. The edge should be traced starting at point (0,0) and going in a counterclockwise direction.

A typical set of information under *2 could be read using the following cards*:

```
*2 1 .002 2 .5394 .5394 .028 3 .2452 .2942 .2942 5 .0984Z7 1 Z 8  .0025
.007 .028 .005 9 .0055 2 Z 10 .0055 2Z 11 4 4 4 4 4 4 4 4 6 Z
12 0 0 0 0
```

Although the entire *2 information is shown in one place, if desired it could have been scattered throughout the input. Further, the values of C can be taken in any desired order, rather than the order taken here. Note that blanks need not be left between Z and a number, but this can be done. In general, the only restriction on the order or quantity of data is that each C group must be complete. If partial input is required and the remainder of the information was read for a previous problem, only the C groups that are to be changed need be read. However, if the change in the C group changes other requirements, then these must be changed as well. An example of this would be a change from one T/E to two. Then new dimensional, material, mesh information, etc. would all be required.

5. <u>*3, Thermoelectric Material Properties</u>

Allowance is made for up to 6 (inclusive) tables of thermoelectric properties. Control numbers, C, are used to indicate the table number (1, 2, 3, ..., 6) and are followed by absolute temperature, seebeck coefficient, thermal

---

* The sample problem is based on information given in MELPAR and the dimensions are given in inches. Any consistent set of dimensions may be used.

conductivity, electrical resistivity, heat capacity, density, and relative emissivity, in the presented order. To read data, first read a control number, then a temperature, then the properties at that temperature, then another temperature, more properties, etc. The end of a table must be indicated by a "Z". This can be followed by another control number if desired, in which case another table can be read. The maximum total number of temperatures that can be read is 200; the minimum number is 3. Linear interpolation is used in the tables.

The temperatures can be read in any desired order, and it is not necessary to number the tables consecutively. If data for one table are to be changed, only that one table need be reread, the others will be unaffected. The same table number may not be read twice during the same reading, but a table number used for a previous problem can be reused for a present problem. In this case the previous table is eliminated.

6. *4, Other Properties

As in the *3 tables, control numbers are utilized to indicate the table (or material) number. The following designations are used:

C	Meaning
7 - 12	Shoe Information
13 - 14	Absorber Information
15 - 16	Radiator (Backup Plate) Information

Information read by C=7 - 12 is a temperature (absolute) and values for thermal conductivity, heat capacity, density, electrical resistivity, and contact resistance (in that order) corresponding to that temperature. "Z's" must be used to indicate the end of each table.

The C=13 - 16 information is identical to C=7 - 12, except a value for relative absorptivity and two values for relative emissivity are read following the electrical value and no contact resistance is read. The relative absorptivity and the first relative emissivity will be applied to the outside (exposed) surface, the second relative emissivity to the internal surface.

Other comments made in the *3 input data section are applicable in *4 data input.

7. *5, Heating Information

This information pertains to the environment as "seen" by the thermoelectric power supply. The following designations for control numbers are used:

C	Meaning
1	Solar constant (heat flux)
2	Planet Reflectivity Constant (albedo)
3	Stefan Boltzmann Constant
4	Time

C	Meaning
5	Mean Planet Temperature as "seen" by the plate
6	Planet view factor for radiant heat transfer from the planet to the absorber
7	Planet view factor for radiant heat transfer from the planet to the radiator
8	Planet view factor for the planet albedo to the absorber
9	Planet view factor for the planet albedo to the radiator
10	Solar angle (Radians)
11	Function Calculation

The first three C's are single-valued constants, and as such need be read only once. The end of these numbers need not be signaled by using a "Z". The other C's are used to designate groups, and the end of each group of numbers must be signaled by a "Z".

If only one value of time (C=4 information) is read, and that time is zero, then a steady state problem will be assumed. If nonzero values are read, then at least three values must be read, and they must be read in increasing order. The lowest value may be zero. Up to 100 (inclusive) values may be read.

The C=5 - 10 information corresponds to the values of time (C=4). If a constant is desired, only one value need be read. Otherwise, the same number of values as used for time (C=4) must be read. The meaning of these numbers can be obtained by considering the following brief discussion.

The solar heat flux absorbed by a flat surface is given by:

$$\begin{aligned} q &= S \cos(\delta)\,\alpha & 0 \le \delta < \frac{\pi}{2} \\ &= 0 & \frac{\pi}{2} \le \delta \le \pi \end{aligned} \qquad \text{(III-1)}$$

where:

q = heat flux
S = solar constant
δ = angle measure from a normal to the surface

In this analysis, $0 \le \delta \le \frac{\pi}{2}$ will be taken to mean that solar heat is impinging on the collector; $\frac{\pi}{2} < \delta = \pi$ will mean it is impinging on the radiator, and $\delta = \frac{\pi}{2}$ means that no solar heat is being absorbed. The heat flux absorbed by the thermoelectric system from a planet is given by:

$$q = \sigma F_1 T^4 \epsilon \qquad \text{(III-2)}$$

where:

σ = Stefan Boltzmann Constant
F_1 = Radiation view factor
T = average planet temperature as seen by the thermoelectric system

The albedo heat flux is given by:

$$q = SRF_2 \alpha \qquad \text{(III-3)}$$

where:

R = Reflectivity Constant (albedo)

Much of this information can be calculated if orbital parameters are defined, and provision is made to do this if desired by using the C=11 input. In this case, subroutine HEAT (TIME, T, FIC, FIR, F2C, F2R, DELTA) will be called. *

8. *6, Other Data

Information read using the *6 designation has the following meanings:

C	Meaning
1	Load resistance
2	Symmetry designation
3	Transient problem investigation time
4	Minimum allowable time increment
5	Maximum allowable number of iterations
6	Maximum allowable temperature change per iteration
7	Steady state convergence factor
8	Frequency of long print-out in transient calculations

* The arguments are the C=4 - 10 variables, and each is a 100 number array. These data should be placed in the array in the same manner as though they were read,except the use of "Z's" is not permitted. Zeros must be placed in all portions of the array not specifically used. The same number of values as times must be used. This subroutine is not presently available, and a dummy subroutine of the same name has been included in the deck.

C	Meaning
9	Frequency of short print-out in transient calculations
10	Mesh specification: If zero or negative, the mesh calculations are to be performed; if equal to one, the mesh calculations are to be skipped.
11	Node map generation designation: If zero, no map is generated in the print-out; if one, a map is generated.
12	Transient-steady state designation: If one, a steady state problem; if zero, a transient problem.
13	Convergence criterion for steady state calculations
14	Frequency of long print-out in terms of number of iterations
15	Frequency of short print-out in terms of number of iterations
16	Estimated number of iterations required for convergence in steady state calculations
17	Collector material number
18	Radiator material number
19	Not assigned
20	System initial temperature

Each of these items is discussed in more detail, where required, in the following paragraphs.

The load resistance, specified by a control number equal to one, is that resistance "seen" by the portion of the solar flat plate being simulated by the computer program.

A control number of two refers to a symmetry designation. Proper use of this term will result in drastic decreases in computation time and a saving of close to a factor of eight can be obtained in many cases. The symmetry designation number is the number of nodes in a repeating portion of the solar flat plate. To see exactly what is meant by this statement, consider Figure III-1. This figure illustrates a completely symmetrical solar flat plate with the thermoelement located in the center and with six nodes specified on each of the four sides. Each of these "rays" is shown in the figure. Specification of a symmetry number of 12 with this configuration would cause the computer program to calculate temperatures beginning at line AO and continuing counterclockwise to line DO. No calculations would be performed for temperatures from line DO counterclockwise to line AO since they would be a mirror image of the previously calculated temperature. Any time a detailed print-out was required, the temperatures from the calculated portion would be reproduced in the uncalculated portion and then all nodal points would be represented in the output.

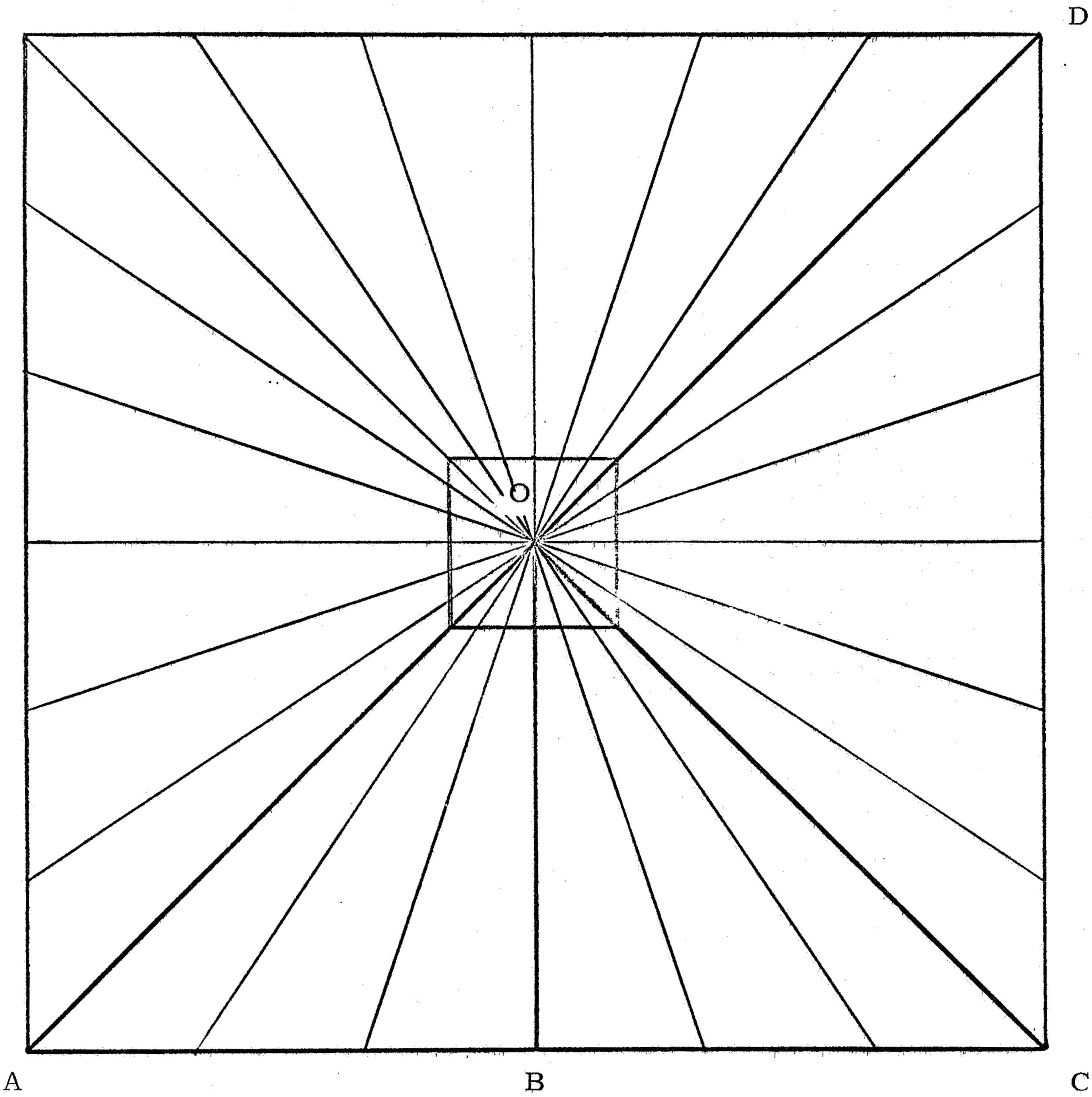


Figure III-1. Illustration of Symmetry Designation Number

Specification of a symmetry number of six would cause calculations to be performed beginning at line AO and progressing in a counterclockwise direction to line CO. No other calculations would be performed, and a detailed print-out would be obtained in a manner similar to that discussed for a symmetry number of 12. Specification of a symmetry number of three would cause calculations to begin at line AO and progress in a counterclockwise direction to line BO. This latter specification would result in a factor of eight less calculations than if no symmetry number were defined or if a symmetry number of 24 were read. The advantage of taking into account the symmetry of the solar flat plate immediately becomes apparent.

Note that a requirement for a symmetrical designation is that the lines of symmetry correspond to nodal separation lines as set up by the computer program. For example, if the number of nodes between points A and C were specified as five rather than six, and similar designations were made for the other three sides, a symmetry specification of five or ten could be made. In this case, a factor of four or two, respectively, decrease in the number of calculations would be obtained. A factor of eight decrease in the number of calculations would be impossible with this designation because integer numbers must be specified, and a specification of 2.5 would be required to take full advantage of the symmetry of the figure, an impossible arrangement.

Care should be taken in the selection of the symmetry number since if a designation is given which is inconsistent with the figure generated by the computer program, the symmetry specification will be neglected and calculations will be performed for all nodes in the solar flat plate. This obviously could cause a considerable increase in running time over that expected with a correct symmetry designation.

A control number of three specifies the maximum time which is to be simulated for investigation of a transient calculation. The problem is terminated when this simulated time has been reached by the computer program. This particular number is not used if a steady state type of calculation has been specified.

A control number of four can be used to represent the minimum time step that is allowable for a transient calculation. The problem will be rejected if a lower time step is required for calculation purposes. A zero is allowable as an input for this number if desired, and a zero will be used if no number is read.

A control number of five defines the maximum number of iterations which will be permitted during a run. If this number is exceeded, the problem will be terminated.

The maximum temperature change per iteration that will be allowed is specified with a control number of six. The calculations will be performed for both the transient and the steady state utilizing this value. In the transient calculation, the time increment will be selected so that the maximum change in temperature is approximately the input value. In the steady state, this value will be the allowable change per iteration, and an extrapolation factor will be selected so that this value is attained (see also the input with a control number of 16).

Input with a control number of seven is the convergence factor utilized for a steady state calculation. A value between 0.5 and 0.8 is recommended (see also the input under a control number of 16).

A control number of eight is used to specify the time separation between detailed temperature print-outs. The temperatures which are outputted will be presented as a function of node number, and ordinarily a node map will be required to interpret the results. This number is ignored for steady state calculations.

The time separation between concise print-outs is specified with a control number of nine. A concise print-out consists of the time, time increment, iteration number, electrical power output, electrical current, the error in the heat balance, the overall conversion efficiency based upon the heat received by the absorber and the electrical output, the hot junction temperature, the cold junction temperature, the average radiator temperature, and the average absorber temperature. A concise print-out is obtained automatically whenever a detailed print-out is output. A long and short print-out is given at the termination of each problem. Specifications with a control number of nine are ignored if steady state calculations are being performed.

A control number of ten is a flag which defines whether the mesh is to be set up or not. The mesh must be calculated if any input has been used in a problem which can change the volume of a node, the separation of one node from another, or the node connection arrangements. If a mesh calculation is specified by reading a zero or negative number with the control number of ten, the temperature specified as an initial condition (*6 20) is used to set the temperature of all of the nodes. If the mesh is not to be calculated, which occurs with a specification of one, then the node temperatures which were previously used represent the starting point of the calculations.

Changes in the following input designations do not require the generation of a new mesh: *1, *2 5, *2 7, *2 9, *2 10, *3, *4, *5, *6 1, *6 2, *6 3, *6 4, *6 5, *6 6, *6 7, *6 8, *6 9, *6 12, *6 13, *6 14, *6 15, *6 16, *6 17, *6 18, and *7. All other input specifications require that a new mesh be generated.

A control number of 11 specifies whether a print-out of the node map is desired or not. If a value of zero is read, no map will be detailed. If a value of one is input, then a complete map will be generated. This map will consist of a scale sketch generated by the computer using the output printer which illustrates the outline of the absorber, and each of the node centers will be located within this sketch to within roughly one part in one hundred. Coordinate numbers will be generated along the sides of this map so that each of the nodes may be referred to by two coordinate numbers. A table will be generated which lists the node numbers corresponding to each of the node centers in terms of the reference numbers generated on the map. The node representations for the radiator will not be generated in a map since the map would be identical; however, the complete representation of these in tabular form will be generated.

The data read using a control number of 13 will be utilized by the computer program to determine when convergence has been achieved in steady state calculations. This number should be the temperature change per iteration that it is desired to attain during steady state calculations. When a maximum temperature change during an iteration has resulted which is less than this number, the problem will be terminated. (Note that in setting up the print specification requirements, final answers are always given upon problem termination.) The number input using a control number of 13 is ignored if a transient calculation is specified.

A control number of 14 causes detailed print-out of results to occur for both the steady state and transient calculations. The value read is the number of iterations between each detailed print-out. If detailed print-outs as a function of iterations are not desired, a large number should be used.

The information obtained by the program using a control number of 15 is identical to that previously presented for 14, except in this case the concise print-out is to be described.

A control number of 16 causes an estimate of the number of steps required for convergence in the steady state to be read. This estimate is used by the computer program during the calculation process. For all calculations performed prior to reaching the number of iteration steps given in this estimate, a change in temperature per iteration as specified using a control number of six is forced. After this estimate has been reached, the temperature change is multiplied by the factor input using a control number of seven in computing temperature changes per iteration. This means that the steady state calculation procedure cannot terminate until after the number of iterations input under the control number 16 has been exceeded. However, the convergence rate will be forced prior to this number being reached. The effectiveness of this "forcing" decreases with the number of iterations performed.

A control number of 17 refers to the material specified for the collector, and the one under a control number of 18 refers to the radiator material. These values are the table numbers which were input using the *4 input.

The initial temperature of the entire generator should be input using a control number of 20. Absolute values of temperature must be used, but other than this restriction, any units desired can be selected, provided consistency is used in all input.

9. *7, *8, *9 Information

These control cards signal the end of input for a particular problem. If *7 is used, calculations will be performed for the data read into the machine. When the required computations have been completed, control will be transferred to read further input information for a new problem. The input for the new problem need contain only those items that have been changed from the previous calculation. All of the previously read information is preserved unless replaced by new input.

The use of a *8 control card will cause behavior similar to the *7 card, except the input for the new problem must be complete. None of the previously read information will be available for a new problem.

The *9 card signals that no input data follows. The required calculations will be performed, and then control will be transferred back to the computer systems tape and the run will be terminated.

B. PROGRAM DESCRIPTION

1. Introduction

This computer program is written for an IBM 7094 which has approximately 32,000 words of storage. Roughly 20,000 words are provided at all times within the core for storage of variables so that the lengthy input-output of data required for calculations is not necessary except to obtain initial information or to provide answers. However, the computer program requires much more memory than the remaining 12,000 words. For this reason, it has been broken into a number of links. The main control of the computation process and those subroutines which require extensive use are always left in the core of the machine. The other subroutines are brought into the core as necessary. The arrangement of subroutines is such that a subroutine need be brought into the core only once for each problem that is to be investigated.

The names of the subroutines and the link to which they are assigned are shown in Table III-2. The overlay construction is shown in Figure III-2.

2. Main Program

The main program serves basically as a calling sequence and controller for all subroutines. It first causes all storage locations used by this program to be cleared. It then causes the input data routines to be selected, selects the routines which process the data and store it in the proper locations, sets up the routines which automatically generate the mesh, and then calls in the basic calculation routine. When calculations have been completed on one problem, it either selects input for a new problem or causes the machine to be returned to the system control.

3. Subroutine LENGTH

This is a very short subroutine which merely computes the straight line distance between two points when the x and y coordinates are given.

4. Subroutine INTER

This subroutine is used to compute the (x, y) coordinates of the point of intersection of two straight lines. The straight lines are described by giving the (x, y) coordinates of two points on each of the lines.

TABLE III-2. COMPUTER PROGRAM OVERLAY CONSTRUCTION

Subroutine Name	Link Number
MAIN	0
POINT	1
LENGTH	1
INTER	1
AREA	1
CHECK	2
PRO-3	3
PRO-5	4
HEAT	4
PRO-4	5
PRO-6	6
PRO-2	6
NUMBER	7
READIT	7
UNPAK	7
PACK	7
SQUASH	7
MESH	8
MESHA	9
MESHB	10
MESHC	10
FPMAP	11
PROP	12
CALCS	13
CALCT	13

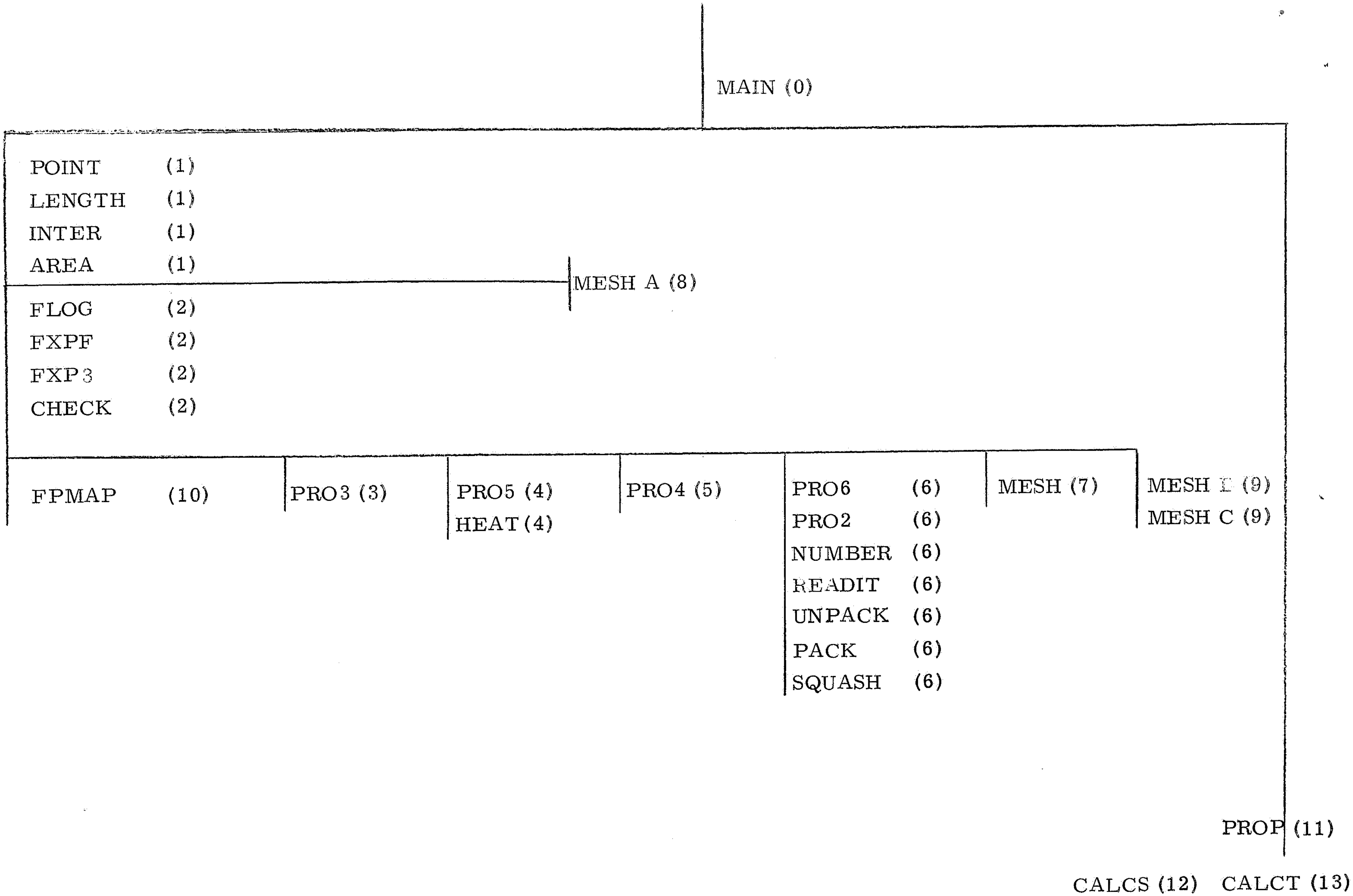


Figure III-2. Overlay Construction

5. Subroutine POINT

The basic calculations are set up using a cylindrical coordinate system. However, it is necessary to relate this to the (x, y) coordinate system which is utilized in providing input to the program. This subroutine takes the radius and angular information being utilized in the program and converts to Cartesian coordinates.

6. Subroutine AREA

Each of the nodes used for calculation purposes is considered as being located at the centroid of its assigned area. This latter value is computed by considering smaller areas of triangular cross section. Subroutine AREA computes the centroid locations in Cartesian coordinates and the surface area from the Cartesian coordinates which describe the three corners of the triangle.

7. Subroutine HEAT

As explained in the input description section, subroutine HEAT is not presently available. However, a dummy subroutine is presently provided. If it is desired in the future to provide automatic calculation of the surroundings as "seen" by the solar flat plate power supply, a suitable subroutine may be added at this point. The actual location of this subroutine in the link structure is relatively unimportant since it will only be called once for each problem. The data it provides is utilized in exactly the same manner as though input data were provided. The calculation process is based upon a linear interpolation of this information.

8. Subroutine NUMBER

This subroutine generates integer or decimal numbers from the components as read by the machine using subroutine READIT. An integer number furnished to this subroutine will consist of only two portions, a sign and a significant figure. Subroutine NUMBER multiplies these two portions, and then stores the result in storage allocations corresponding to the type 2, 3, 4, 5, or 6 input.

A decimal or floating point number consists of several portions. The exponent consists of a sign and a significant figure portion, and the significant figure portions consist of a sign, the integer (that portion of the number to the left of the decimal point), and the significant figures which occur to the right of the decimal point. Subroutine NUMBER compiles these various pieces to form the completed number and then stores them in the proper location as previously described for an integer number.

9. Subroutine READIT

All input to this program is by A-mode conversion. This enables the program to read any recognized FORTRAN character. Subroutine READIT causes these data to be read and then processes the information.

The initial character on an input card is first checked to see if it has an asterisk, and if an asterisk is found, the character is column 2 is shifted to the right-most position in its storage location so that the type of control card may be determined. If an illegal character is found in column 2, the entire card is printed out with a comment to the effect that an illegal control card was found, and the program is removed from the computer. If the first card of data read is found not to be a control card, it is assumed that the control card is missing and the problem will be refused. If a legitimate control card is found, it is analyzed to see if this is information pertaining to input data, or if the information defines the end of a problem. If an end of problem is found, control is transferred back to the MAIN PROGRAM for the beginning of the processing of the information which has been read and stored in the machine. If the control card indicates that data are to be processed, then the required action is taken.

If the card read is a comment card (*1), then the first two columns of the card are considered to be blank, and the remainder of the card is printed as output. If the card is determined to contain numerical information, then that information is processed. Each of the allowed characters - BLANK, -, +, ·, E, Z, 0, or 1 through 9 - is checked and appropriate action is taken.

Construction of a number first begins with a sign indication, or if no sign indication is found, with an integer portion. If a BLANK is encountered during the compilation of the integer, the end of the number is assumed, and subroutine NUMBER is called. Any other allowed character will either be a single digit integer, a decimal point, or exponent type information. An integer is treated by simply shifting the previous integer to the left and adding the new one. A decimal point merely sets a flag so that further integers are stored in a different location and are treated as the significant figure portion to the right of the decimal point. Any of several characters may indicate an exponent and if this is found, a flag is set. Any further information obtained after the flag is set, but before a BLANK is found, is incorporated into the exponential portion of the number prior to calling subroutine NUMBER which processes the information and stores it in the right location.

If at any time an illegal character or an unallowed number is found, or a return is received from subroutine NUMBER with an error designation, then a check is made to determine if any further problems remain. If no problems remain, the program is automatically removed from the computer. If further problems remain, then a flag is set and subroutine READIT transfers control back to the MAIN PROGRAM. The MAIN PROGRAM is caused to clear the portions of the machine allotted for data storage, set a flag, and reenter subroutine READIT. This institutes a search process of the remaining data until a complete new problem is found. At this point, analysis will proceed.

10. Subroutine UNPAK

One of the basic ground rules under which this computer program was written was that the input-output of information from the computer core would be limited to input data and output answers. The initial processing of input information required the storage of a large number of one or two digit integer numbers. This type of information requires only a few bits in a word, and the remainder of the word is not used. Consequently, when a shortage of core storage was encountered, the integer numbers were compressed so that eight were stored in a single word location in the IBM 7094 memory. Subroutine PACK causes this compression of the integer numbers to take place. Subroutine UNPAK causes the inverse to occur.

These subroutines require a small MAP routine (the equivalent of the FORTRAN IV machine language). The two entry points of this machine language portion are titled SQUASH and SPREAD. Their functions are self-explanatory.

11. Subroutine CHECK

There is no way of determining the difference between a minus zero and a plus zero using FORTRAN IV. Consequently, a small MAP subroutine was written which provides a solution to the problem.

12. Subroutine PRO-2

This subroutine is called after all of the information has been processed by READIT. This particular routine processes all of the information that has been input using the *2 control. The subroutine causes all of the data to be processed and stored in the necessary memory allocations which have been assigned for each of the variables. All of the input data are cross-checked to be certain that sufficient data have been provided and that contradictory or inconsistent data have been excluded. The first of several checks on the number of mesh points specified is also made in this subroutine. If an error is found, a flag is set and control is returned to the MAIN PROGRAM for processing of the remaining information to determine if another problem remains which can be attempted.

13. Subroutine PRO-3

The function of this subroutine is quite similar to PRO-2 except that the type of information to be stored is different. This subroutine first causes all of the thermoelectric information that has been previously stored in the computer memory to be moved to a different location for temporary storage. The new information is then processed and all of the property data are stored in the designated locations. It is possible at this point that the data could have been input to the computer in a nonascending temperature order, and consequently the data are immediately reprocessed so that all information is in the order of increasing temperature. After this has been done, checks are made for specifications of duplicate table numbers on the input data. If none are found, then the information which was saved from previous problems is merged with the new information so that a complete set of data are available. If, during this merge process, any duplicate tables are encountered, the old tables are eliminated.

Once all of the tabular information has been compiled and sorted, a check is made of the information which will be required during the calculation process. If it is found that information is going to be requested which is unavailable, a comment will be made to this effect and an error return procedure will be instigated. Otherwise, notations are set up so that the information can be used during the calculation process.

14. Subroutine PRO-4

This calculation process is virtually identical to that previously discussed in subroutine PRO-3. The only differences are slight perturbations in the material which is to be stored and in the references which are set up so that the data can be later utilized.

15. Subroutine PRO-5

This subroutine is relatively straightforward and short, and serves to merely store the input data in the proper locations. It also contains the calling sequence for subroutine HEAT.

16. Subroutine PRO-6

This subroutine processes the remaining data which are necessary for the utilization of the computer program. In each case, each piece of data information is preceded by a control number which is then utilized in subroutine PRO-6 for storage information purposes.

17. Subroutine MESH

This subroutine contains the beginning of the automatic mesh generation procedure. The first items which are computed are the angles interconnecting the center of the thermoelement with its corners. and the center of the thermoelement with the corners of the attached plate. The number of points specified for the edges of the plate and thermoelement are next investigated to determine the more stringent requirements. When this has been obtained, the previously calculated angles are divided up into small segments. The segments are next smoothed by an iterative process so that the accuracy of the calculations will be improved.

Once these calculations have been done, it is possible to compute the radii corresponding to each of the angular segments at the intersection of the thermoelement and the plate, and at the plate edge. These are stored at the proper locations and then the angular segments are integrated to provide the angle between a line connecting the center of the thermoelement with the Cartesian coordinate point (0, 0), and the angular segment. When this calculation has been completed, and an average angular segment size has been determined for calculation purposes, control is transferred back to the MAIN PROGRAM which then immediately calls subroutine MESHA.

18. Subroutine MESHA

This subroutine continues the calculations which were begun in subroutine MESH. It first causes the stored locations assigned for the areas, the centroids, and the inside and back length to be cleared. It then selects the first node and assigns a trapezoid to the node. The Cartesian coordinates of each of the trapezoid corners are assigned and each of the corners is investigated to see whether it lies within the plate, within the portion of the plate assigned to the thermoelement, or outside of the plate. An index is set up as each of these determinations is made. This index is next used with a computed "GO TO" statement so that the proper breakdown of the trapezoid into triangles for calculation of area and centroids is obtained. (See Section II-C.)

During this calculation process, several nodes will be obtained which are totally within the area assigned to the thermoelement, and others will be obtained which are totally outside of the plate. Suitable indices are set up so that these nodes are not referred to or incorporated into the calculation process. Further, several logical calculations are made to assure that the memory allocation assigned for the storage of node information is not exceeded.

When the calculations in this subroutine have been satisfactorily completed, transfer is made back to the MAIN PROGRAM which then calls in subroutine MESHB.

19. Subroutine MESHB

A number of very small nodes can be readily obtained during the previously described calculation process. These nodes, which can result from portions of the trapezoid either being within the thermoelement or outside of the plate, will contribute nothing to the calculation, and indeed could considerably slow the calculations. Consequently, they are combined in this subroutine with adjoining nodes in such a manner that accuracy is for practical purposes not compromised. This joining process is made by comparing the size of a node with its nearest neighbor. If a factor of four difference in the size is obtained, and certain other criteria are also satisfied, then the node is combined with its neighbor. This combination process consists of the addition of the areas, and the weighted relocation of the node centroid in the Cartesian coordinate system. Information must also be incorporated for the logical calculation of the cross sectional area common to one node and its neighbors.

When this information has been compiled, transfer is made back to the MAIN PROGRAM which then calls in subroutine MESHC.

20. Subroutine MESHC

This subroutine computes the area-to-length ratios necessary for the future calculation of conductances between the various nodes. It also sets up the initial mesh temperatures and finishes setting up the boundary information which is required.

21. Subroutine FPMAP

This subroutine generates a scaled map of the mesh arrangement which has been generated by the computer program. The map is caused to be printed on output paper to a precision of approximately one part in a hundred. Tabular information which will later enable the user to relate temperatures to specific positions is also presented.

22. Subroutine PROP

This subroutine provides linear interpolation of all input data which is either a function of temperature or a function of time. It is utilized during the calculation process so that the physical properties of the absorber, radiator, and thermoelement are treated as temperature dependent variables. Transient characteristics as generated by subroutine HEAT or as described in the input are also processed in this subroutine.

23. Subroutine CALCS

This subroutine constitutes the main steady state calculation process of the program and is the portion of the program where most of the steady state calculation time is expended. It is broken into three portions: mesh property evaluation, steady state calculation, and print-out of results.

The properties of the absorber, radiator, and thermoelements are determined by calling subroutine PROP. These properties are utilized to compute conductances between each node and its interconnected nodes, the heat capacity of the node, and the surroundings information (such as solar input, albedo, planet input, and heat rejection from the exposed surfaces). When these data have been computed, a number of other characteristics which will be necessary are computed and stored (such as temperature to the fourth power).

The calculation is begun by first considering the hot junction node. The conductance for this node to all of the surrounding nodes is obtained by integrating with respect to angle through 360° (or less if a symmetry specification has been incorporated). The conductance to the thermoelement is next computed, and the various effects such as Peltier cooling and contact resistance heating are included. The calculation of a new temperature immediately follows.

The calculation of the cold end node temperature is virtually identical. Once this has been computed, the thermoelement calculations are made on a node-by-node basis, and then the absorber calculations are made. The same calculations are utilized for the absorber as for the radiator, and the radiator is computed by merely resetting some indices and using the same equations.

Once new temperatures have been obtained, they are multiplied by the extrapolation factor and checks are made to determine whether further calculations must be performed.

The generation of print-out information is relatively straightforward and will not be discussed in detail.

24. Subroutine CALCT

This subroutine performs the transient calculations for the program, and most of the calculation time for this type of problem will be spent here. The routine is very similar to subroutine CALCS, discussed in the previous subsection, and the only basic difference is the addition of heat capacity and time terms to the steady state calculation. The calculation process is virtually identical and need not be discussed further.

IV. CALCULATIONS PERFORMED USING THE PROGRAM

A. Sample Problem

The input data for a sample program and the resulting computer output are presented in the following pages. Figure IV-1 shows sample input data for the calculation of the characteristics of a repeating section of the present solar flat plate configuration. This specific problem did not contain a request for a map of the array. The array which would have been generated is shown in Figure IV-2 which was obtained from a previous computer run. Figure IV-3 gives map coordinates, x - y coordinates, and the node number. Also shown are the summary results when convergence was obtained. For this steady state problem, the time and time increment are meaningless. The iteration number indicates that it took 2456 iterations for the specified convergence to be obtained. The power output from the repeating section was 0.00566 watts, the current was 0.373 amps, and the fractional error in the answer obtained by comparing with an overall heat balance was 0.00259. The overall efficiency based upon the heat absorbed by the panel is 2.63 percent. The hot junction temperature computed was 922°R, and the cold junction temperature 636°R. These compare to an average absorber temperature of 937°R and an average radiator temperature of 624°R. These average temperatures were obtained by a weighting of temperature to the fourth power so that the true heat rejection characteristics of the surfaces are obtained. The computer printout which gives the detailed temperature distribution is also shown in Figure IV-3. The location of each of the presented temperatures may be obtained by referring to the previously generated node numbers.

Similar results are obtained if the node specification is changed to three or four nodes on a side. The node map for a three node specification is shown in Figure IV-4 and Table IV-1, that for a four node specification in Figure IV-5 and Table IV-2.

B. Comparison with Previous Calculations

Fuschillo* has presented an analysis of the Solar Flat Plate Generator using manual calculation techniques. Since this is the most complete analysis which has been published, it will be used as a basis for comparison.

---

*Fuschillo, N., R. Gibson, F.K. Eggleston and J. Epstein, "Flat Plate Solar Thermoelectric Generator for Near-Earth Orbits," Advanced Energy Conversion, Vol. 6, pp. 103-125, 1966.

```
SS BASE CASE
*2 1 .002 2 .5394 .5394 .028 3 .2452 .2452 .2942 .2942 5 .0984Z7 1 Z
8 .0025 .028 .005 9 .0055 7 Z 10 .C055 7Z 11 2 2 2 2 2 2 2 2 6
Z 12 0 0 0 0
*3 1 200 1.C6-4 1.56E-2 .996E-3 35.5 .277 .1
500. 1.06E-4 1.56E-2 .996E-3 35.5 .277 .1
1000 1.06-4 1.56E-2 .996E-3 35.5 .277 .1  Z
*4 7 200 100 20 20 .1 0
500 100 20 20 .1 0
1000 100 20 20 .1 0 Z
13 200 3.22 229. .1  .1 .82 .08 .03
500    3.22 229. .1  .1 .82 .08 .03
1000   3.22 229. .1  .1 .82 .08 .03Z
15 200 3.22 229. .1  .1 .14 .95 .03
500    3.22 229. .1  .1 .14 .95 .03
1000   3.22 229. .1  .1 .14 .95 .03 Z
*5 1 .903 2 .0 3 .00349E-9 4 0 Z 5  0 Z 6 .094 Z 7 .094 Z 10  0 Z 8
 .155 Z 9 .155 Z
*6 1 4.07-2 2 1 3 150  4 1-5 5 1+5 6 1. 7 .7 8 5. 9 .5 10 0 11 0
12 1 13 1-3 14 500 15 50 16 2000 17 13 18 15 19 0 20 700
*9
```

Figure IV-1. Sample Input

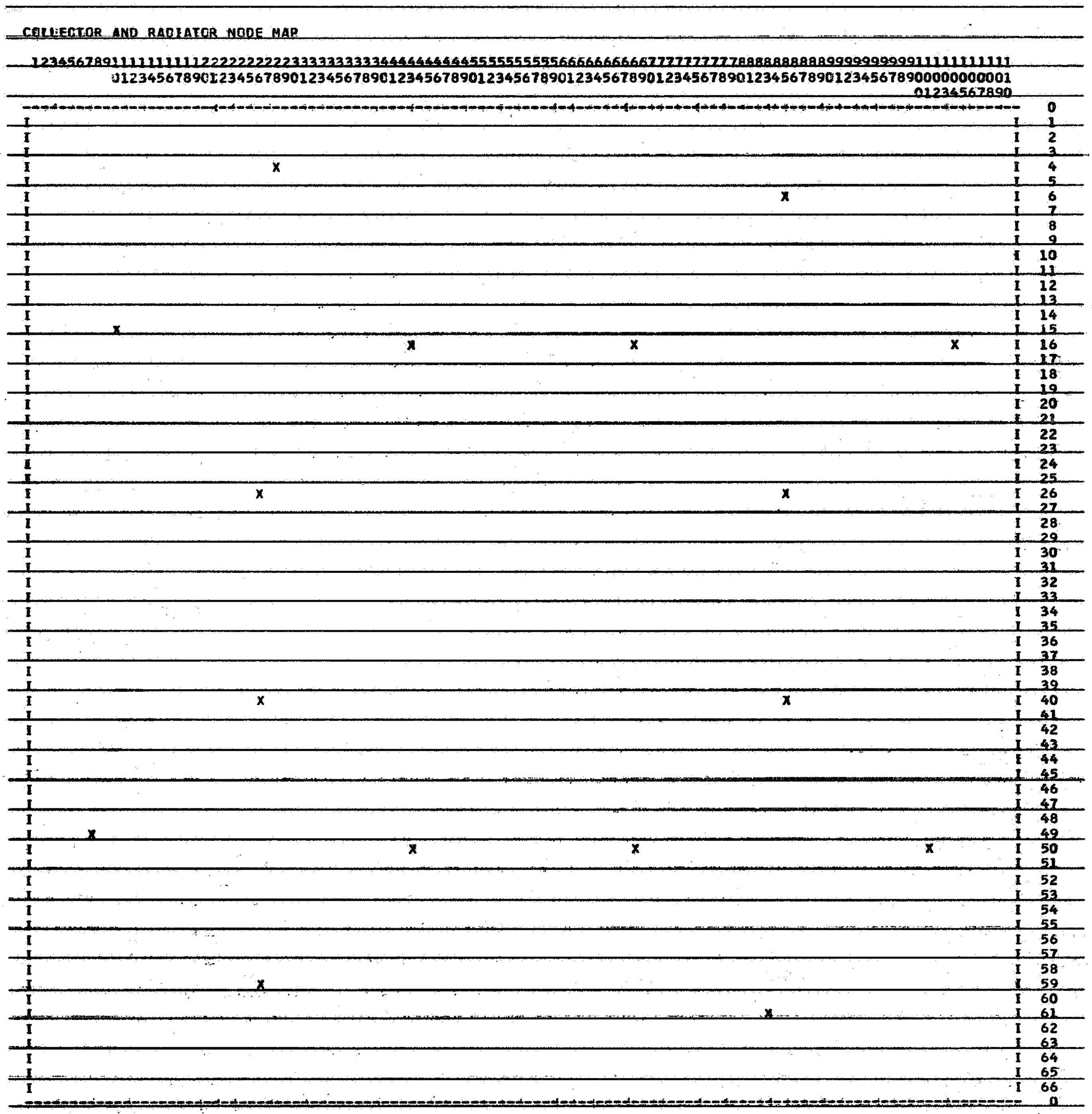


Figure IV-2. Node Map

MAP COORDINATES		X-Y COORDINATES		NODE NUMBER
28	4	0.1325E-00	0.5073E 00	18
85	6	0.4142E-00	0.4906E-00	16
10	15	0.4876E-01	0.4142E-00	20
43	16	0.2103E-00	0.4131E-00	17
68	16	0.3291E-00	0.4131E-00	15
104	16	0.5073E 00	0.4069E-00	14
26	26	0.1263E-00	0.3291E-00	19
85	26	0.4131E-00	0.3291E-00	13
26	40	0.1263E-00	0.2103E-00	21
85	40	0.4131E-00	0.2103E-00	11
7	49	0.3210E-01	0.1325E-00	22
43	50	0.2103E-00	0.1263E-00	7
68	50	0.3291E-00	0.1263E-00	9
101	50	0.4906E-00	0.1252E-00	12
26	59	0.1252E-00	0.4876E-01	8
83	61	0.4069E-00	0.3210E-01	10

TIME	0.
TIME INCREMENT	1.000000
ITERATION NO.	2456.0
POWER	0.00566
CURRENT	0.37278
ERROR	0.002588
EFFICIENCY	2.625324
TEMPERATURES	
HOT JUNCTION	922.3103
COLD JUNCTION	635.6225
RADIATOR (AVG)	624.2953
ABSORBER (AVG)	937.0653

THERMOELEMENT MESH INFORMATION

NODE NUMBER	X-Y-Z COORDINATES		
1	0.26970E-00	0.26970E-00	0.82000E-02
2	0.26970E-00	0.26970E-00	0.24600E-01
3	0.26970E-00	0.26970E-00	0.41000E-01
4	0.26970E-00	0.26970E-00	0.57400E-01
5	0.26970E-00	0.26970E-00	0.73800E-01
6	0.26970E-00	0.26970E-00	0.90200E-01

TEMPERATURES

NODE NUMBER	0	1	2	3	4	5	6	7	8	9
0		660.25	708.95	757.13	804.84	852.12	899.02	936.80	937.47	936.80
10	937.47	936.80	937.47	936.80	937.47	936.80	937.47	936.80	937.47	936.80
20	937.47	936.80	937.47	922.31	624.50	623.99	624.50	623.99	624.50	623.99
30	624.50	623.99	624.50	623.99	624.50	623.99	624.50	623.99	624.50	623.99
40	635.62	0.	0.	0.	0.	0.	0.	0.	0.	0.

Figure IV-3. Computed Results

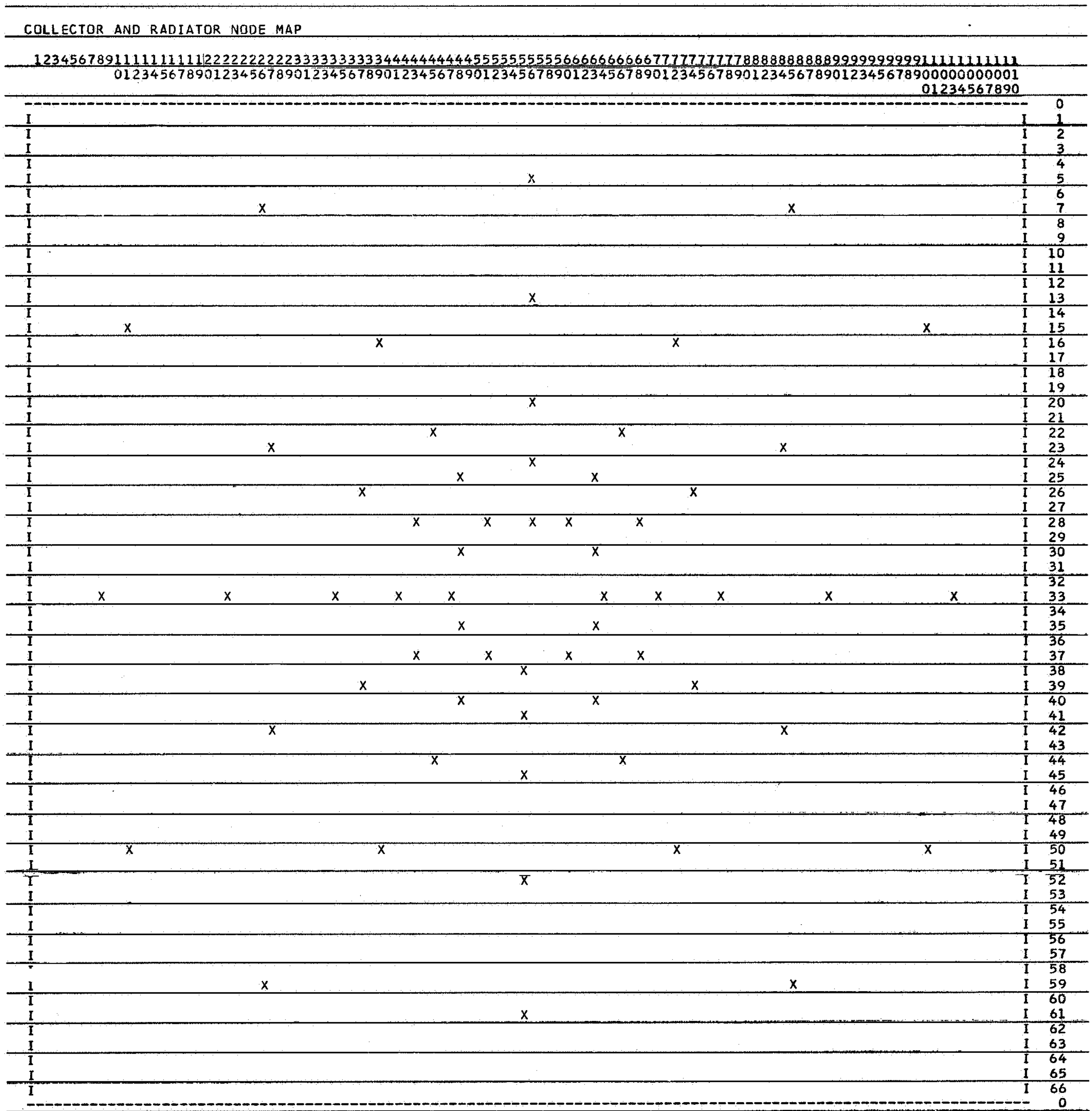


Figure IV-4. Three Node Map

```
 COLLECTOR AND RADIATOR NODE MAP

 12345678911111111112222222222333333333344444444445555555555666666666677777777778888888888999999999911111111111
          01234567890123456789012345678901234567890123456789012345678901234567890123456789012345678900000000001
                                                                                                    01234567890
----------------------------------------------------------------------------------------------------------------   0
I                                                                                                              I   1
I                                           X                      X                                           I   2
I                                                                                                              I   3
I                                                                                                              I   4
I                   X                                                                      X                   I   5
I                                                                                                              I   6
I                                                                                                              I   7
I                                              X                X                                              I   8
I                                                                                                              I   9
I                                                                                                              I  10
I                                                                                                              I  11
I       X                       X                                              X                       X       I  12
I                                                                                                              I  13
I                                                                                                              I  14
I                                                X            X                                                I  15
I                                                                                                              I  16
I                                                                                                              I  17
I                                     X                                  X                                     I  18
I                   X                                                                      X                   I  19
I                                                  X        X                                                  I  20
I                                                                                                              I  21
I                                          X                        X                                          I  22
I                             X                     X      X                     X                             I  23
I                                                                                                              I  24
I                                    X         X                X         X                                    I  25
I   X                                                X    X                                                X   I  26
I                                         X      X            X      X                                         I  27
I            X                                        X  X                                        X            I  28
I                        X                    X    X        X    X                    X                        I  29
I                                 X              X            X              X                                 I  30
I                                       X   X                      X   X                                       I  31
I                                               X              X                                               I  32
I                                                                                                              I  33
I                                       X   X   X              X   X   X                                       I  34
I                                 X              X            X              X                                 I  35
I                        X                    X             X    X                    X                        I  36
I                                                  X  X  X                                                     I  37
I            X                            X      X            X      X                            X            I  38
I                                                    X    X                                                    I  39
I   X                                X         X                X         X                                X   I  40
I                                                                                                              I  41
I                                                   X      X                                                   I  42
I                             X            X                        X            X                             I  43
I                                                                                                              I  44
I                                                  X        X                                                  I  45
I                                                                                                              I  46
I                   X                                                                      X                   I  47
I                                     X                                  X                                     I  48
I                                                                                                              I  49
I                                                X            X                                                I  50
I                                                                                                              I  51
I                                                                                                              I  52
I       X                                                                                              X       I  53
I                               X                                              X                               I  54
I                                                                                                              I  55
I                                                                                                              I  56
I                                              X                X                                              I  57
I                                                                                                              I  58
I                                                                                                              I  59
I                                                                                                              I  60
I                   X                                                                      X                   I  61
I                                                                                                              I  62
I                                           X                      X                                           I  63
I                                                                                                              I  64
I                                                                                                              I  65
I                                                                                                              I  66
----------------------------------------------------------------------------------------------------------------   0
```

Figure IV-5. Four Node Map

MAP COORDINATES		X-Y COORDINATES		Node NUMBER
56	5	0.2697E-00	0.5037E 00	46
26	7	0.1247E-00	0.4888E-00	51
85	7	0.4147E-00	0.4887E-00	41
56	13	0.2697E-00	0.4329E-00	45
11	15	0.5067E-01	0.4147E-00	56
100	15	0.4887E-00	0.4147E-00	36
39	16	0.1881E-00	0.4111E-00	50
72	16	0.3513E-00	0.4111E-00	40
56	20	0.2697E-00	0.3768E-00	44
45	22	0.2161E-00	0.3625E-00	49
66	22	0.3233E-00	0.3625E-00	39
27	23	0.1283E-00	0.3513E-00	55
84	23	0.4111E-00	0.3513E-00	35
56	24	0.2697E-00	0.3400E-00	43
48	25	0.2345E-00	0.3306E-00	48
63	25	0.3049E-00	0.3306E-00	38
37	26	0.1769E-00	0.3233E-00	54
74	26	0.3625E-00	0.3233E-00	34
43	28	0.2088E-00	0.3049E-00	53
51	28	0.2484E-00	0.3072E-00	47
56	28	0.2697E-00	0.3114E-00	42
60	28	0.2911E-00	0.3071E-00	37
68	28	0.3306E-00	0.3049E-00	33
48	30	0.2323E-00	0.2911E-00	52
63	30	0.3072E-00	0.2910E-00	32
8	33	0.3566E-01	0.2697E-00	61
22	33	0.1065E-00	0.2697E-00	60
34	33	0.1626E-00	0.2697E-00	59
41	33	0.1994E-00	0.2697E-00	58
47	33	0.2280E-00	0.2697E-00	57
64	33	0.3114E-00	0.2697E-00	27
70	33	0.3400E-00	0.2697E-00	28
77	33	0.3768E-00	0.2697E-00	29

MAP COORDINATES		X-Y COORDINATES		Node NUMBER
89	33	0.4329E-00	0.2697E-00	30
103	33	0.5037E 00	0.2697E-00	31
48	35	0.2322E-00	0.2484E-00	62
63	35	0.3071E-00	0.2483E-00	22
43	37	0.2088E-00	0.2345E-00	63
51	37	0.2483E-00	0.2323E-00	7
60	37	0.2910E-00	0.2322E-00	17
68	37	0.3306E-00	0.2345E-00	23
55	38	0.2697E-00	0.2280E-00	12
37	39	0.1769E-00	0.2161E-00	64
74	39	0.3625E-00	0.2161E-00	24
48	40	0.2345E-00	0.2088E-00	8
63	40	0.3049E-00	0.2088E-00	18
55	41	0.2697E-00	0.1994E-00	13
27	42	0.1283E-00	0.1881E-00	65
84	42	0.4111E-00	0.1881E-00	25
45	44	0.2161E-00	0.1769E-00	9
66	44	0.3233E-00	0.1769E-00	19
55	45	0.2697E-00	0.1626E-00	14
11	50	0.5067E-01	0.1247E-00	66
39	50	0.1881E-00	0.1283E-00	10
72	50	0.3513E-00	0.1283E-00	20
100	50	0.4887E-00	0.1247E-00	26
55	52	0.2697E-00	0.1065E-00	15
26	59	0.1247E-00	0.5067E-01	11
85	59	0.4147E-00	0.5066E-01	21
55	61	0.2697E-00	0.3566E-01	16

THEMOELEMENT MESH INFORMATION

NODE NUMBER	X-Y-Z COORDINATES		
1	0.26970E-00	0.26970E-00	0.82000E-02
2	0.26970E-00	0.26970E-00	0.24600E-01
3	0.26970E-00	0.26970E-00	0.41000E-01
4	0.26970E-00	0.26970E-00	0.57400E-01
5	0.26970E-00	0.26970E-00	0.73800E-01
6	0.26970E-00	0.26970E-00	0.90200E-01

Table IV-1. Three Node Specification

MAP COORDINATES		X-Y COORDINATES		Node NUMBER
44	2	0.2122E-00	0.5240E 00	83
67	2	0.3272E-00	0.5240E 00	76
20	5	0.9640E-01	0.5051E 00	90
91	5	0.4430E-00	0.5051E 00	69
47	8	0.2284E-00	0.4773E-00	82
64	8	0.3110E-00	0.4773E-00	75
8	12	0.3434E-01	0.4430E-00	97
32	12	0.1521E-00	0.4457E-00	89
79	12	0.3873E-00	0.4457E-00	68
103	12	0.5051E 00	0.4430E-00	62
49	15	0.2401E-00	0.4188E-00	81
62	15	0.2993E-00	0.4188E-00	74
38	18	0.1853E-00	0.3961E-00	88
73	18	0.3541E-00	0.3961E-00	67
20	19	0.9372E-01	0.3873E-00	96
91	19	0.4457E-00	0.3873E-00	61
51	20	0.2484E-00	0.3767E-00	80
60	20	0.2910E-00	0.3767E-00	73
43	22	0.2091E-00	0.3604E-00	87
68	22	0.3303E-00	0.3604E-00	66
30	23	0.1433E-00	0.3541E-00	95
52	23	0.2544E-00	0.3465E-00	79
59	23	0.2850E-00	0.3465E-00	72
81	23	0.3961E-00	0.3541E-00	60
37	25	0.1790E-00	0.3303E-00	94
47	25	0.2262E-00	0.3348E-00	86
64	25	0.3132E-00	0.3348E-00	65
74	25	0.3604E-00	0.3303E-00	59
4	26	0.1535E-01	0.3272E-00	104
53	26	0.2587E-00	0.3249E-00	78
58	26	0.2807E-00	0.3249E-00	71
107	26	0.5241E 00	0.3272E-00	55
42	27	0.2046E-00	0.3132E-00	93
49	27	0.2384E-00	0.3165E-00	85
62	27	0.3010E-00	0.3165E-00	64
69	27	0.3349E-00	0.3132E-00	58
13	28	0.6211E-01	0.3110E-00	103
54	28	0.2626E-00	0.3059E-00	77

MAP COORDINATES		X-Y COORDINATES		Node NUMBER
57	28	0.2768E-00	0.3059E-00	70
98	28	0.4773E-00	0.3110E-00	54
25	29	0.1206E-00	0.2993E-00	102
46	29	0.2229E-00	0.3010E-00	92
51	29	0.2485E-00	0.3017E-00	84
60	29	0.2906E-00	0.3020E-00	63
65	29	0.3165E-00	0.3010E-00	57
86	29	0.4188E-00	0.2993E-00	53
34	30	0.1627E-00	0.2910E-00	101
49	30	0.2374E-00	0.2906E-00	91
62	30	0.3017E-00	0.2909E-00	56
77	30	0.3767E-00	0.2910E-00	52
40	31	0.1929E-00	0.2850E-00	100
44	31	0.2145E-00	0.2807E-00	99
67	31	0.3249E-00	0.2807E-00	50
71	31	0.3465E-00	0.2850E-00	51
48	32	0.2335E-00	0.2768E-00	98
63	32	0.3059E-00	0.2768E-00	49
40	34	0.1929E-00	0.2544E-00	107
44	34	0.2145E-00	0.2587E-00	106
48	34	0.2336E-00	0.2625E-00	105
63	34	0.3059E-00	0.2626E-00	42
67	34	0.3249E-00	0.2587E-00	43
71	34	0.3465E-00	0.2544E-00	44
34	35	0.1627E-00	0.2484E-00	108
49	35	0.2377E-00	0.2485E-00	112
62	35	0.3020E-00	0.2488E-00	35
77	35	0.3767E-00	0.2484E-00	45
25	36	0.1206E-00	0.2401E-00	109
46	36	0.2229E-00	0.2384E-00	113
60	36	0.2909E-00	0.2377E-00	28
65	36	0.3165E-00	0.2384E-00	36
86	36	0.4188E-00	0.2401E-00	46
51	37	0.2488E-00	0.2374E-00	7
54	37	0.2626E-00	0.2335E-00	14
57	37	0.2769E-00	0.2336E-00	21

Table IV-2. Four Node Specification

map coordinates		x-y coordinates		node number
13	38	0.6211E-01	0.2284E-00	110
42	38	0.2045E-00	0.2262E-00	114
49	38	0.2384E-00	0.2229E-00	8
62	38	0.3010E-00	0.2229E-00	29
69	38	0.3348E-00	0.2262E-00	37
98	38	0.4773E-00	0.2284E-00	47
53	39	0.2587E-00	0.2145E-00	15
58	39	0.2807E-00	0.2145E-00	22
4	40	0.1535E-01	0.2122E-00	111
37	40	0.1790E-00	0.2091E-00	115
47	40	0.2262E-00	0.2046E-00	9
64	40	0.3132E-00	0.2046E-00	30
74	40	0.3604E-00	0.2091E-00	38
107	40	0.5240E 00	0.2122E-00	48
52	42	0.2544E-00	0.1929E-00	16
59	42	0.2850E-00	0.1929E-00	23
30	43	0.1433E-00	0.1853E-00	116
43	43	0.2091E-00	0.1790E-00	10
68	43	0.3303E-00	0.1790E-00	31
81	43	0.3961E-00	0.1853E-00	39
51	45	0.2484E-00	0.1627E-00	17
60	45	0.2910E-00	0.1627E-00	24
20	47	0.9372E-01	0.1521E-00	117
91	47	0.4457E-00	0.1521E-00	40
38	48	0.1853E-00	0.1433E-00	11
73	48	0.3541E-00	0.1433E-00	32
49	50	0.2401E-00	0.1206E-00	18
62	50	0.2993E-00	0.1206E-00	25
8	53	0.3434E-01	0.9641E-01	118
103	53	0.5051E 00	0.9641E-01	41
32	54	0.1521E-00	0.9372E-01	12
79	54	0.3873E-00	0.9372E-01	33
47	57	0.2284E-00	0.6211E-01	19
64	57	0.3110E-00	0.6211E-01	26
20	61	0.9641E-01	0.3434E-01	13
91	61	0.4430E-00	0.3434E-01	34
44	63	0.2122E-00	0.1535E-01	20
67	63	0.3272E-00	0.1535E-01	27

THEMOELEMENT MESH INFORMATION

NODE NUMBER	X-Y-Z COORDINATES		
1	0.26970E-00	0.26970E-00	0.82000E-02
2	0.26970E-00	0.26970E-00	0.24600E-01
3	0.26970E-00	0.26970E-00	0.41000E-01
4	0.26970E-00	0.26970E-00	0.57400E-01
5	0.26970E-00	0.26970E-00	0.73800E-01
6	0.26970E-00	0.26970E-00	0.90200E-01

Table IV-2 Continued. Four Node Specification

An examination of the data presented by Fuschillo shows that thermal conductivity data are not presented for the thermoelements analyzed. Further, use of the thermoelectric properties presented in his Figures 8 and 9 is not always in agreement with the mean property values utilized for the calculations. Consequently, to provide the same basis between the calculations, the data have been obtained by "back calculations" from his results.

A design point is selected by Fuschillo on page 112 for the manufactured solar flat plate panels. They are stated to have the following characteristics:

Hot Junction Temperature (T_h)	520^oK
Absorber Relative Absorbtivity (α_a)	0.82
Absorber Relative Emissivity (ϵ_a)	0.08
Cold Junction Temperature (T_c)	350^oK
Radiator Relative Emissivity (ϵ_r)	0.95
Average Figure of Merit ($\overline{Z}$)	$1.3 \times 10^{-3}\ {}^oK^{-1}$
ZT_h	0.67
Power Output	3.3 watts/ft^2
Plate Area (A) to Length (ℓ) Ratio	30 cm
Open Circuit Voltage ($V_o = S\Delta T$, S = Seebeck Coefficient)	64×10^{-3} Volts
Internal Plate Relative Emissivity	0.03

Examination of these data shows that Fuschillo's Figure 7, line D is applicable. This yields a value of $\overline{k}A/\ell = 0.31$ where $\overline{k}$ is the mean thermoelement thermal conductivity. This immediately allows the thermal conductivity of the thermoelement to be calculated. Since $\overline{Z}$ and $\overline{k}$ are now known, and by definition:

$$\overline{Z} = \frac{\overline{S}^2}{\overline{\rho}\ \overline{k}}$$

where $\overline{S}$ = Seebeck voltage; $\overline{\rho}$ = electrical resistivity

We find:

$$\frac{\overline{S}^2}{\overline{\rho}} = \overline{k}\ \ \overline{Z} = 1.440 \times 10^{-5}\ \frac{\text{watts}}{\text{cm}^2\text{-}{}^oK^2}$$

From Fuschillo, page 112, for these temperature conditions:

$$V_o = S\,\Delta T = 64 \times 10^{-3}\ \text{v.}$$

where V_o = open circuit voltage; $\Delta T = T_h - T_c$

$$S = \frac{64 \times 10^{-3}}{520-350} = 3.82 \times 10^{-4} \text{ v}/^{\circ}\text{K}$$

Now:

$$\bar{S} = \frac{S}{2} = \frac{3.82 \times 10^{-4}}{2} = 1.91 \times 10^{-4} \text{ v}/^{\circ}\text{K}$$

$$\bar{\rho} = \frac{(1.91)^2(10^{-8})}{(1.44)(10^{-5})} = 2.53 \times 10^{-3} \ \Omega \text{ cm}$$

The length of the thermoelements is 0.25 cm. An area to length ratio of thirty (30) yields a corresponding plate side dimension of 2.74 cm. Since the repeating section analyzed by the code is one fourth of the area, or one half of the side dimension, the repeating section analyzed is 1.37 cm which is very close to the value presented in Fuschillo's Figure 11 for the thermoelements which have been constructed and which are analyzed herein.

Fuschillo's analysis, as shown in his Figure 1, is based upon a load resistance equal to the internal thermoelement resistance. Since the tested thermoelement is 0.25 cm long and has a cross sectional area of $(0.1245)^2$ cm^2, the resistance is:

$$R = \frac{(0.25)(2.53)(10^{-3})}{(0.1245)^2} = 4.07 \times 10^{-2} \ \Omega$$

These data have been utilized in a computer run with the code modified to hold the hot and cold junction temperatures at 520°K and 350°K, respectively. The thermoelectric behavior predicted by Fuschillo was identical to that predicted by the code under these conditions.

The full calculational capabilities of the code were used with the same input data. These results were presented as the sample problem in Section IV-A. A comparison of the calculated temperatures and the results from Fuschillo's Figure 2 is shown in Table IV-3. It is interesting that the computer results for the average collector temperature and Fuschillo's calculations are virtually identical. The computer predicts an average radiator temperature that is 3°K lower, a hot junction temperature that is 8°K lower, and a cold junction temperature that is 3°K higher. The effect is essentially due to temperature gradient in the radiator and collector, which is to be expected. Since the average collector temperature is by definition that temperature which represents the overall collector from a radiant heat rejection standpoint, the agreement between the manual and computer calculations is to be expected. The temperature decrease to 512°K at the hot junction and the increase to 353°K at the cold junction results in a lower temperature differential across the thermoelement. This in turn causes a six to seven percent decrease in the power output. The three degree difference in the average radiator temperature is probably due to the cross sectional area employed. A difference between the actual number used by Fuschillo and that utilized herein of one percent in the linear side dimension would account for this difference. It would also account for the slightly higher average collector temperature.

Table IV-3 Comparison of Computed Temperature Data

Item	Fuschillo	Code
Hot Junction Temperature, ^{o}K	520	512.39
Cold Junction Temperature, ^{o}K	350	353.12
Average Radiator Temperature, ^{o}K	350	346.83
Average Collector Temperature, ^{o}K	520	520.61

C. Transient Behavior

A transient investigation was conducted to estimate the time required for the solar flat plate to reach its steady state value and also to investigate how rapidly the power dropped when the solar input was removed. The heat up curve shown in Figure IV-6 indicates that the power output is to within 90 percent of the steady state value in 100 seconds and is approaching the steady state value at 200 seconds. The cool down curve indicates that the power output drops extremely rapidly when the solar flux is removed and is approximately one half of the steady state value within 15 seconds.

D. Parametric Investigation With Constant Properties

Variation of the design parameters is of particular interest to this program because of its extensive calculational capability. The data presented in this section are based upon the bismuth telluride data which have been utilized for the previously presented calculations.

The effective changes in the collector and radiator thickness are indicated in Figure IV-7. The present configuration is considered to have a collector thickness of 0.002 in. and a radiator thickness including the effect of the stiffening rib of 0.0025 in. This operating point is indicated in the Figure. The power output shown is for one repeating section of the plate. The efficiency is based upon the electrical power produced at a useful load compared to the energy intercepted by the plate. As would be expected, the effect of increasing collector thickness or radiator thickness

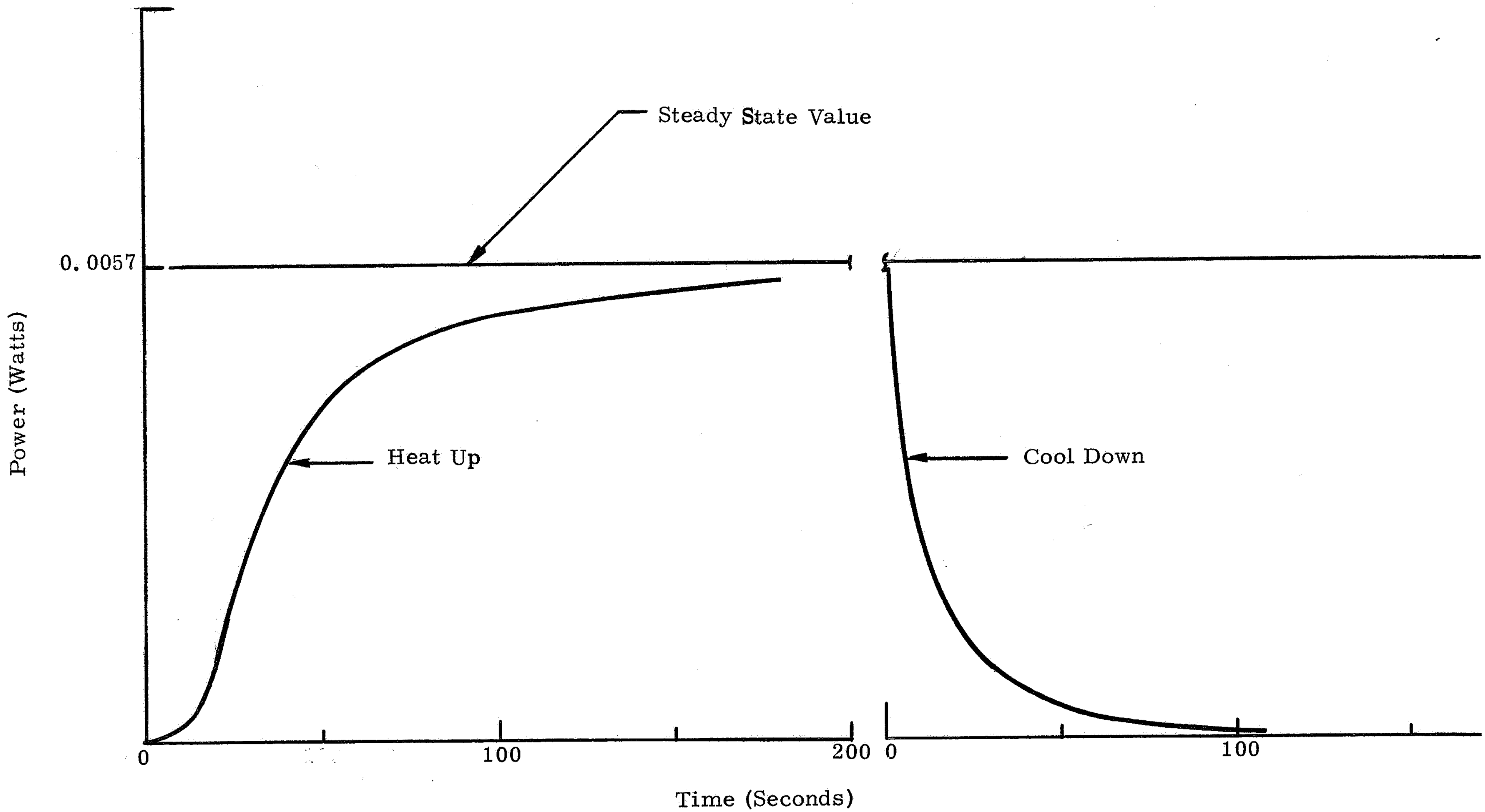


Figure IV-6. Time Transient Investigation

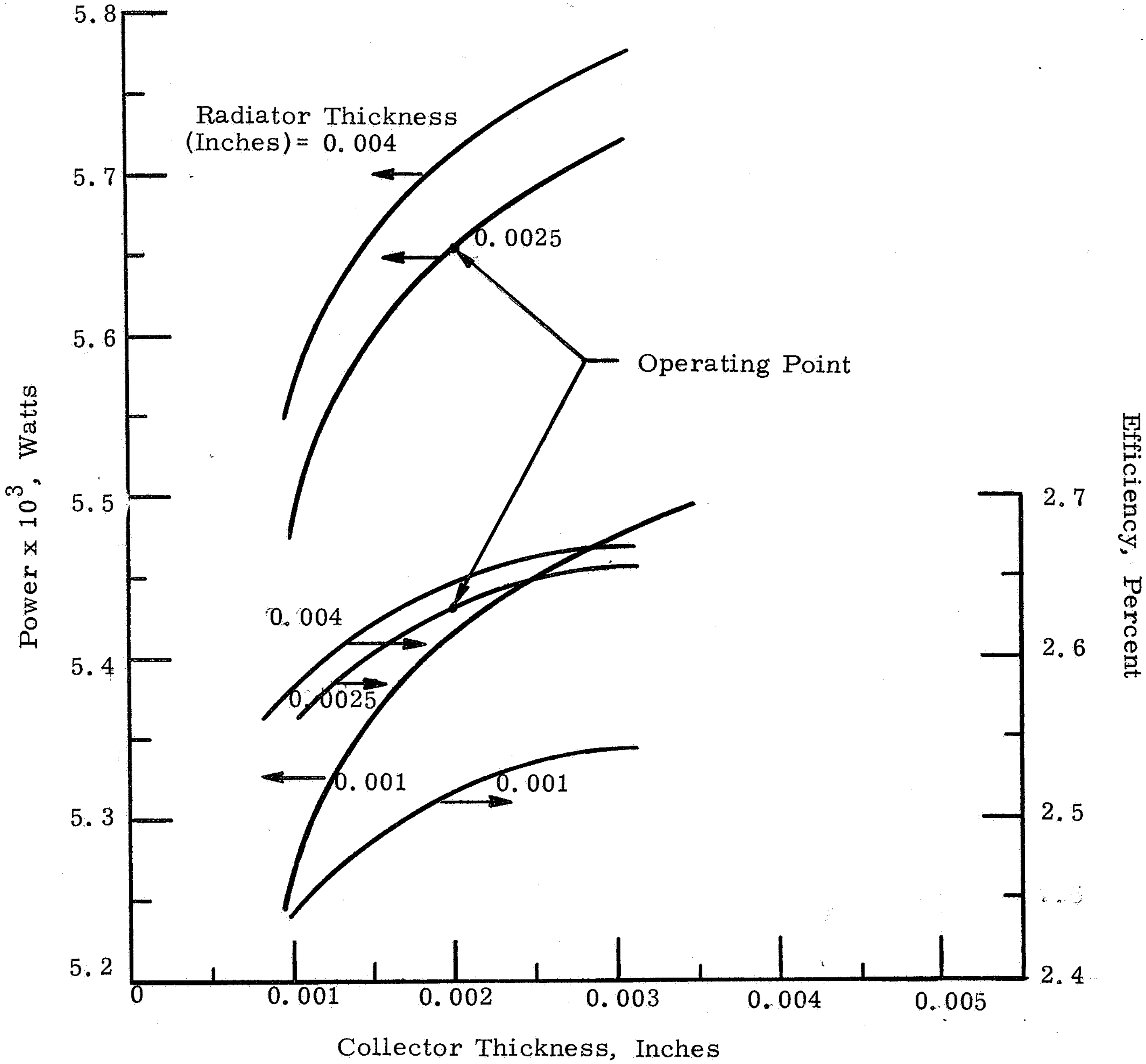


Figure IV-7. Effect of Changes in Plate Thickness is to increase the power output as well as the efficiency. The effect is relatively flat with respect to thickness for small changes (the scales in Figure IV-7 may be somewhat misleading). However, a change in thickness of both collector and radiator from 0.001 in. to 0.004 in. results in a ten percent increase in the power output, which indicates that larger changes are not negligible.

The effect of changing the thermoelement length with all other parameters constant is indicated in Figure IV-8. This figure immediately shows that the overall efficiency of the configuration will increase as the thermoelement length is increased. This particular effect could not be observed utilizing the approach taken by Fuschillo because of his assumption that the internal load resistances were equal. It is known that the best efficiency for conversion of heat to electricity with thermoelements occurs when the load resistance is approximately twenty to forty percent greater than the internal resistance of the thermoelements. This occurs because of the secondary effect of Peltier cooling which is increased by increasing current. Figure IV-8 indicates that the present thermoelements are not an optimum configuration with respect to the thermoelement area to length ratio since the efficiency and power are increasing as the thermoelement length increased. Further, the load resistance utilized for this combination is also not an optimum. This would indicate that further analysis should be employed to generate a more optimum configuration.

The effect of changing the load resistance is shown in Figure IV-9. The present operating point occurs with a load resistance of 0.0407 ohms. This figure indicates that the power output and efficiency will increase with increasing load resistance with a maximum in the vicinity of 0.05 ohms. This value is in the range of the optimum ratio of external to internal load which would normally be expected. The effect of contact resistance or other internal resistances is also indicated in Figure IV-9. This indicates the need for minimizing such resistances, and also strongly emphasizes the need for including them in calculations if realistic answers are to be obtained. It perhaps should be emphasized that all previous answers presented in this analysis have assumed a zero contact resistance and have additionally assumed no wiring resistance.

E. Parametric Investigations Conducted With Variable Bismuth-Telluride Properties

Bismuth-telluride properties should not be represented as constants for parametric studies, particularly when the temperature can change drastically, because several of the variables are immediately being ignored. Typical bismuth-telluride data are shown in Figures IV-10 through IV-12. The thermal conductivity information principally is based upon estimates. These data have been utilized in several additional parametric studies.

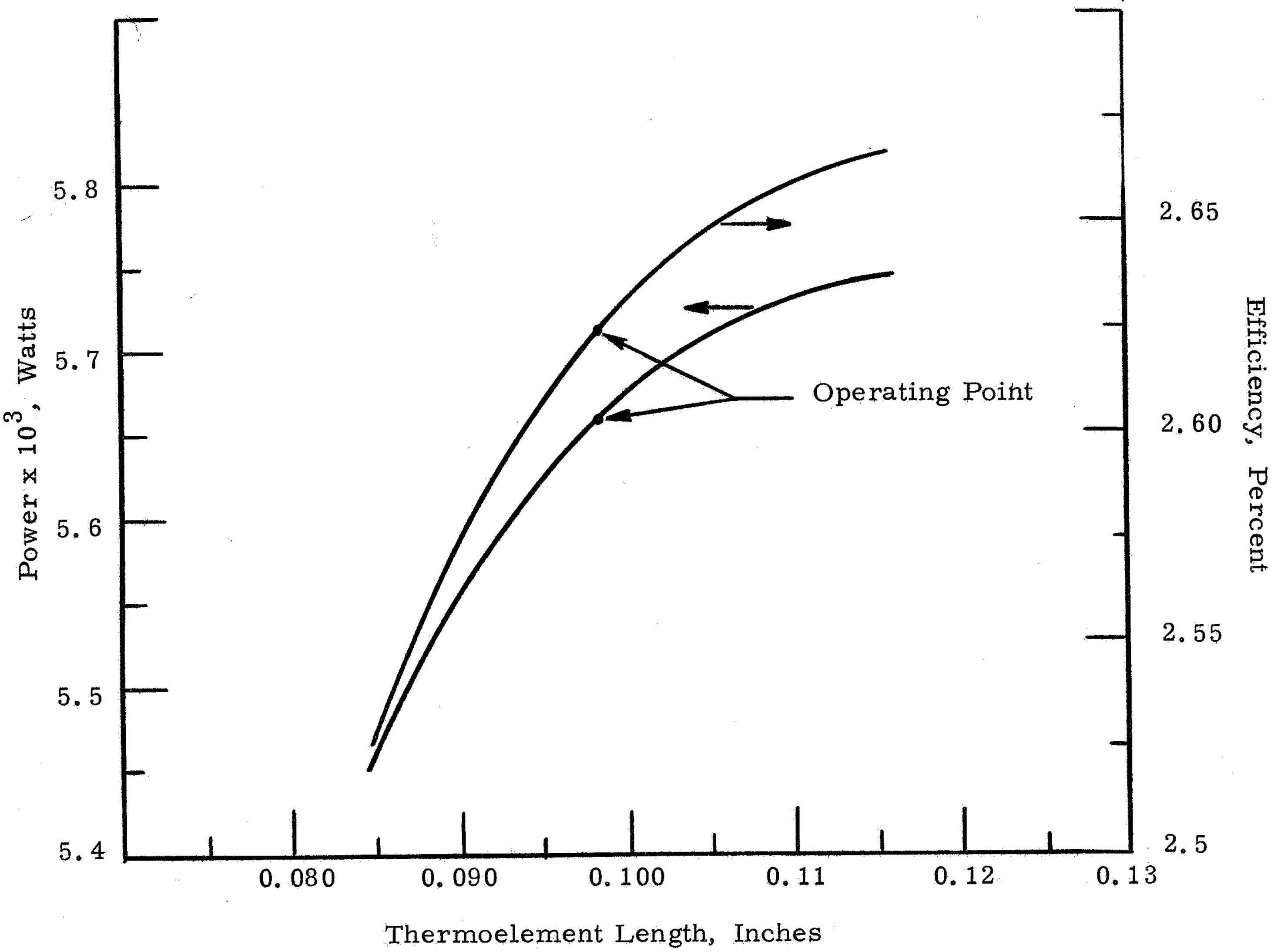


Figure IV-8. The Effect of Changing Thermoelement Length With a Load Resistance of 0.0407 Ohms

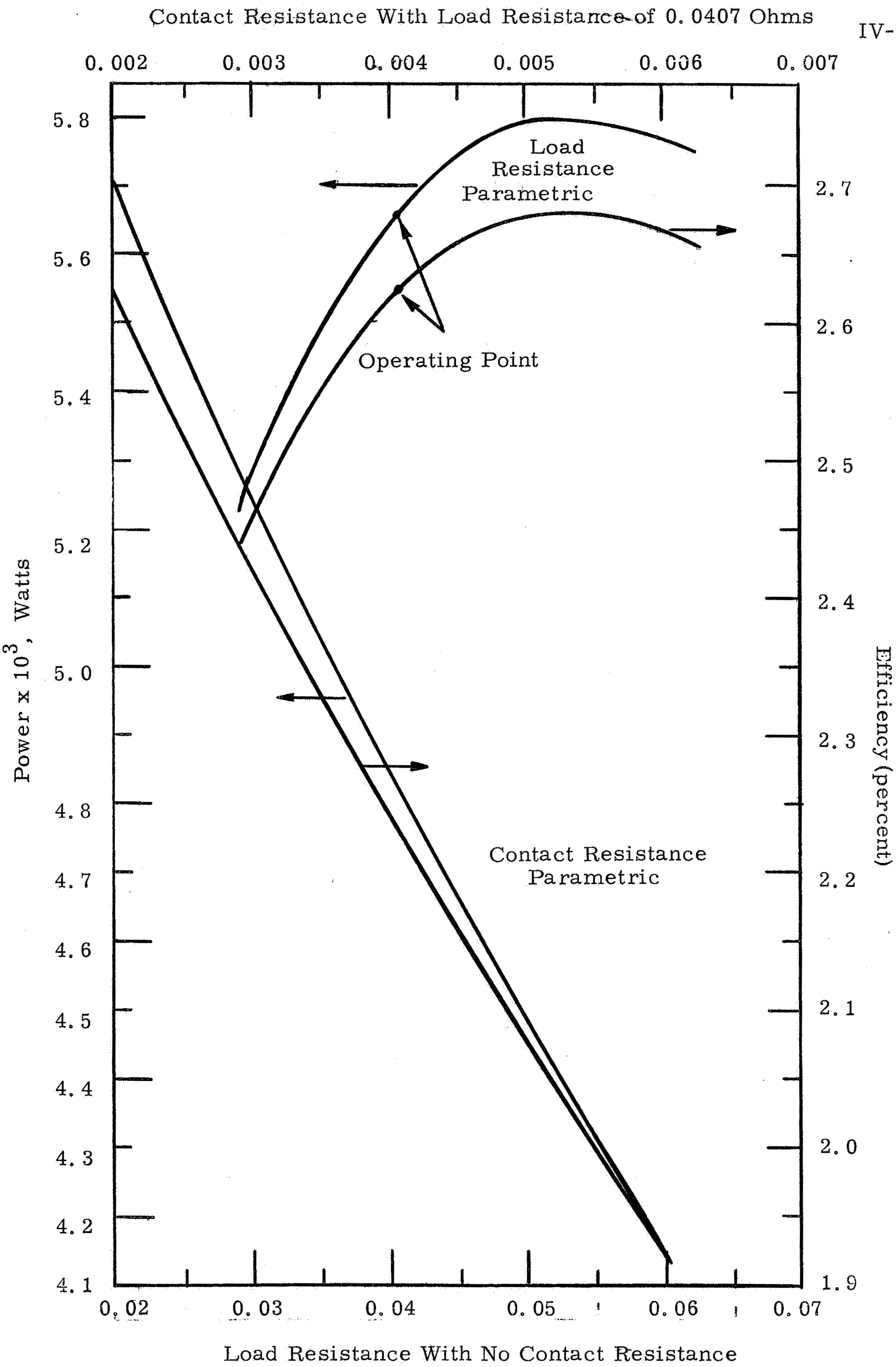


Figure IV-9. Effect of Load and Contact Resistance

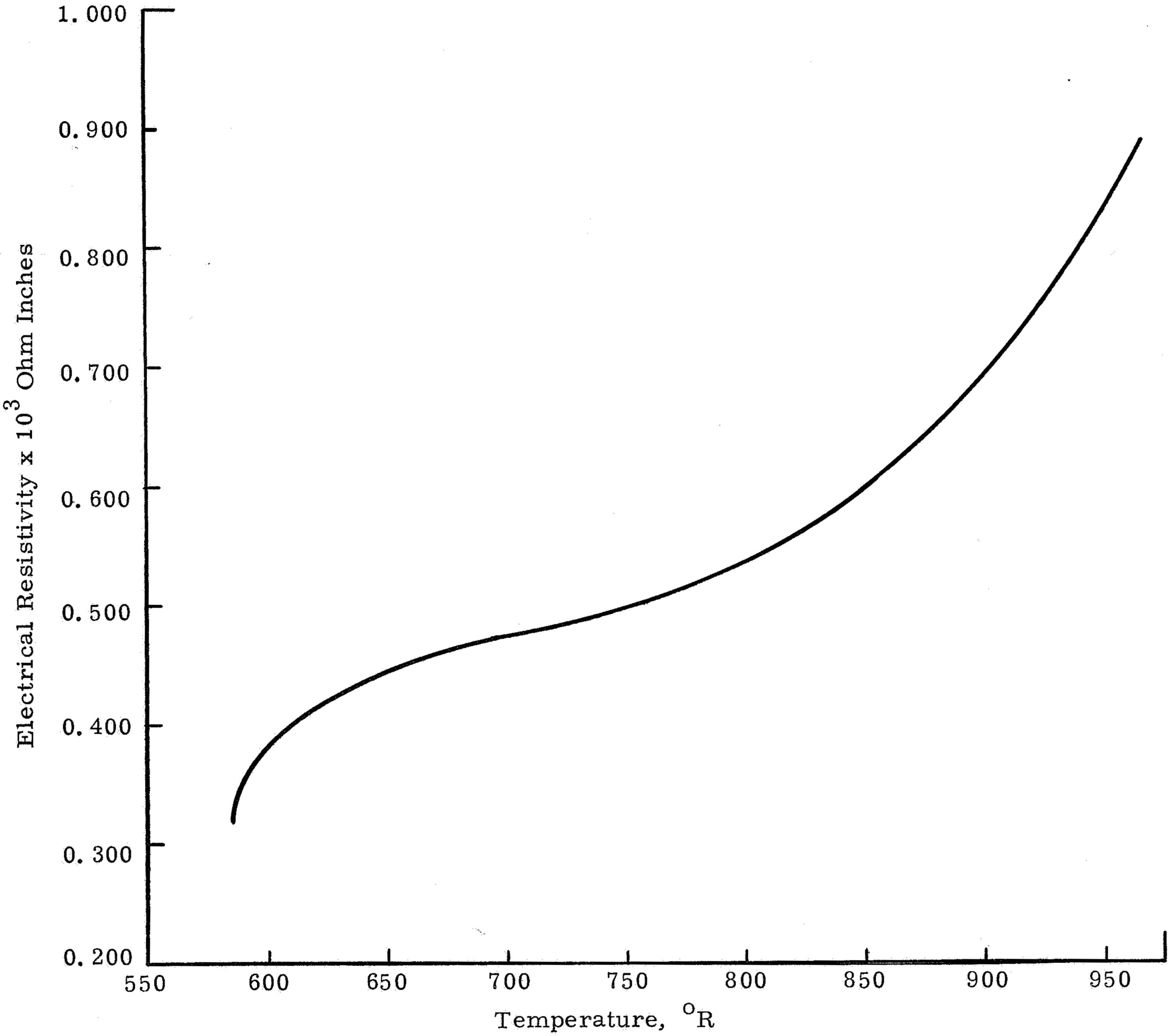


Figure IV-10. BiTe Electrical Resistivity

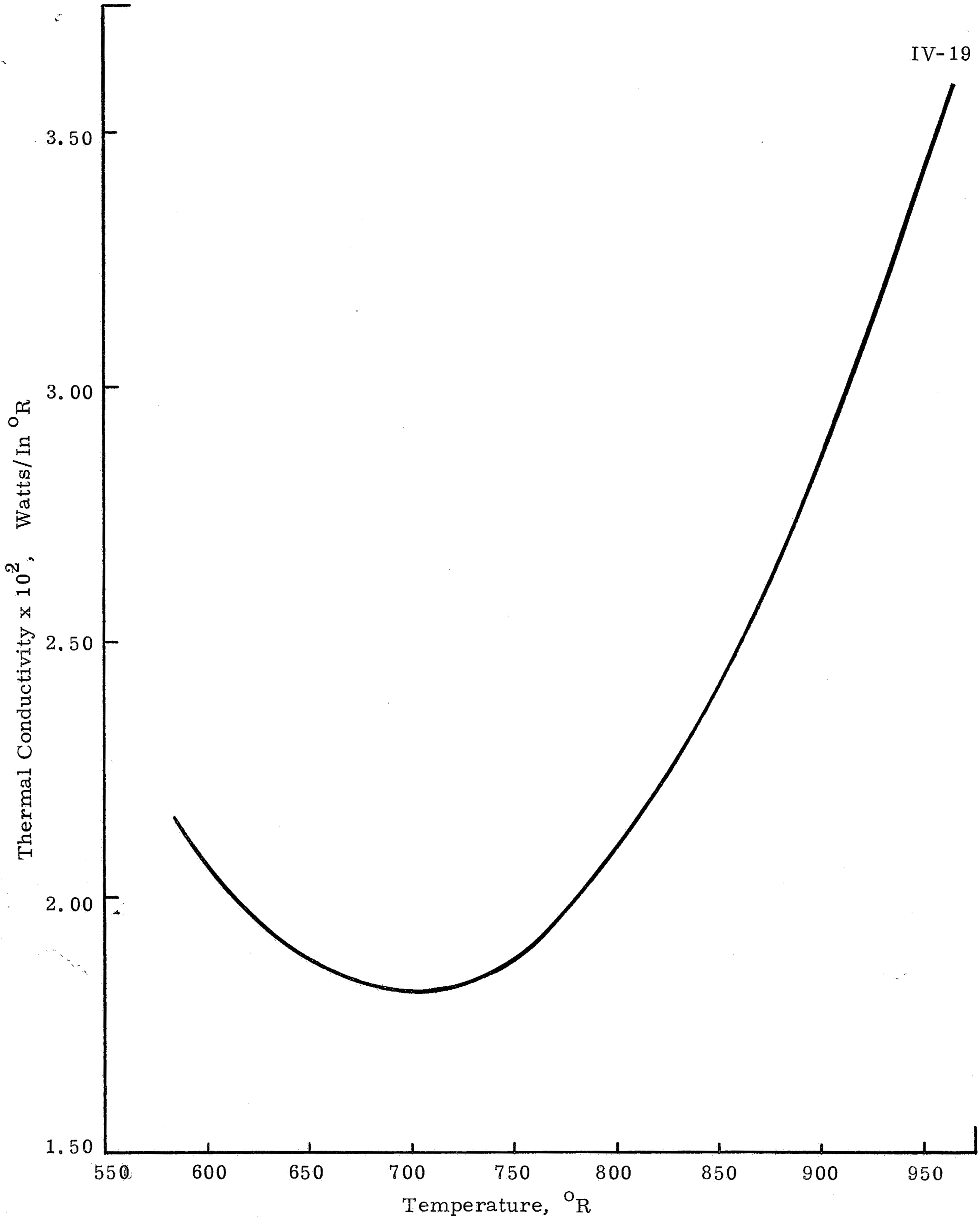


Figure IV-11. BiTe Thermal Conductivity

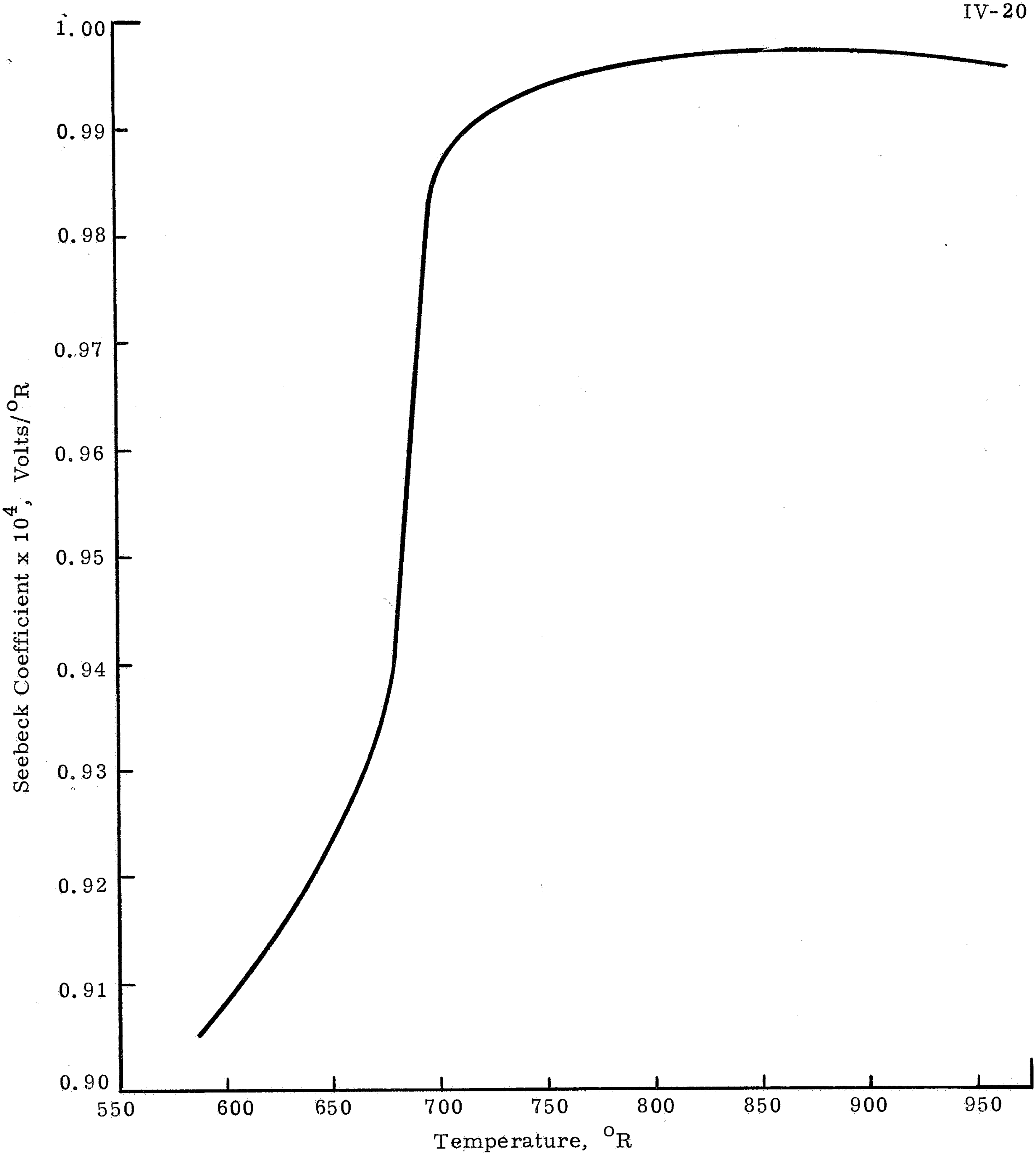


Figure IV-12. BiTe Seebeck Coefficient

The effect of variations in the collector and radiator thickness is shown in Figure IV-13. The reason for the lower power output of these runs as compared to the previously presented information is due to the data used, a mismatch in the load resistance, and the configuration being further removed from optimum than was previously the case. This effect is further indicated in Figure IV-14 which indicates the effect of changing thermoelement length. The steepness of the power output curve with respect to thermoelement length indicates the distance that the indicated design point is removed from the optimum point.

The effect of changing load resistance and contact resistance with these data is indicated in Figure IV-15. The conclusions previously mentioned in regard to contact and wiring resistance remain unchanged. However, here it is seen that the power output, and hence the efficiency, is decreasing with increasing load resistance. This is taking place because the resistivity of the bismuth-telluride used for these calculations was lower than the previous values, and the optimum load resistance is on the opposite side of the operating point.

The effect of changing the plate area while holding other independent values constant is shown in Figure IV-16. As would be expected, the power input increases with increasing surface area because this allows the collector to run hotter while the radiator runs cooler. Of particular interest, however, is the effect of changing the area on power density. Here, it is seen that a maximum is obtained as the area is properly matched to the thermoelement characteristics. This immediately indicates the importance of considering the variation of plate area on the power density. It is further expected that the effect of plate thickness has a rather large effect on the optimum area because of the temperature gradient which would occur.

These parametric studies show that to realize the best performance it is necessary to optimize with respect to thermoelement length, thermoelement cross sectional area, load resistance, and plate area. Further, the effect of plate thickness will have a perturbing effect on the conclusions. In general, the thicker the plate, the more attractive will be the performance, but the heavier will be the resulting solar panel.

F. Solar Flat Plate Using Silicon-Germanium Thermoelements

Application of the solar flat plate to produce electrical power for missions close to the sun is particularly attractive if high temperature thermoelements are utilized. Bifano* has reported results for several investigations at varying distances from the sun. Two of his investigations have been selected for comparison purposes, at 0.388 and 0.25AU from the sun. Typical characteristics are presented in Table IV-4. Since he did not present silicon germanium data, data for this material was obtained from

---

*Bifano, William J., "Analysis of Solar Thermoelectric Flat-Plate Generators Operating in the Range of 1.0 to 0.1 Astronomical Unit", E-3478, Lewis Research Center, NASA, Cleveland, Ohio.

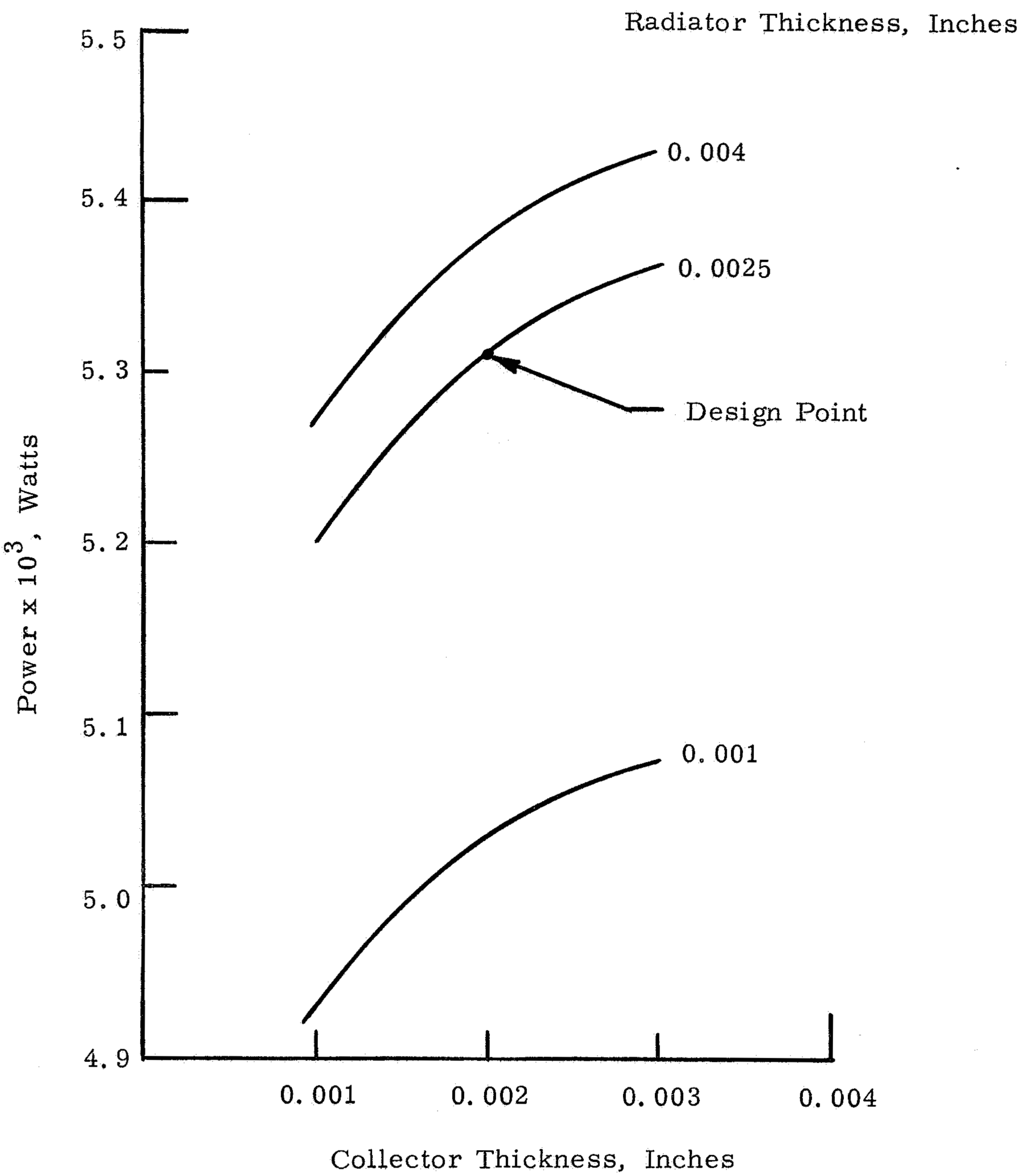


Figure IV-13. Thickness Parametric With Variable Bi-Te Properties

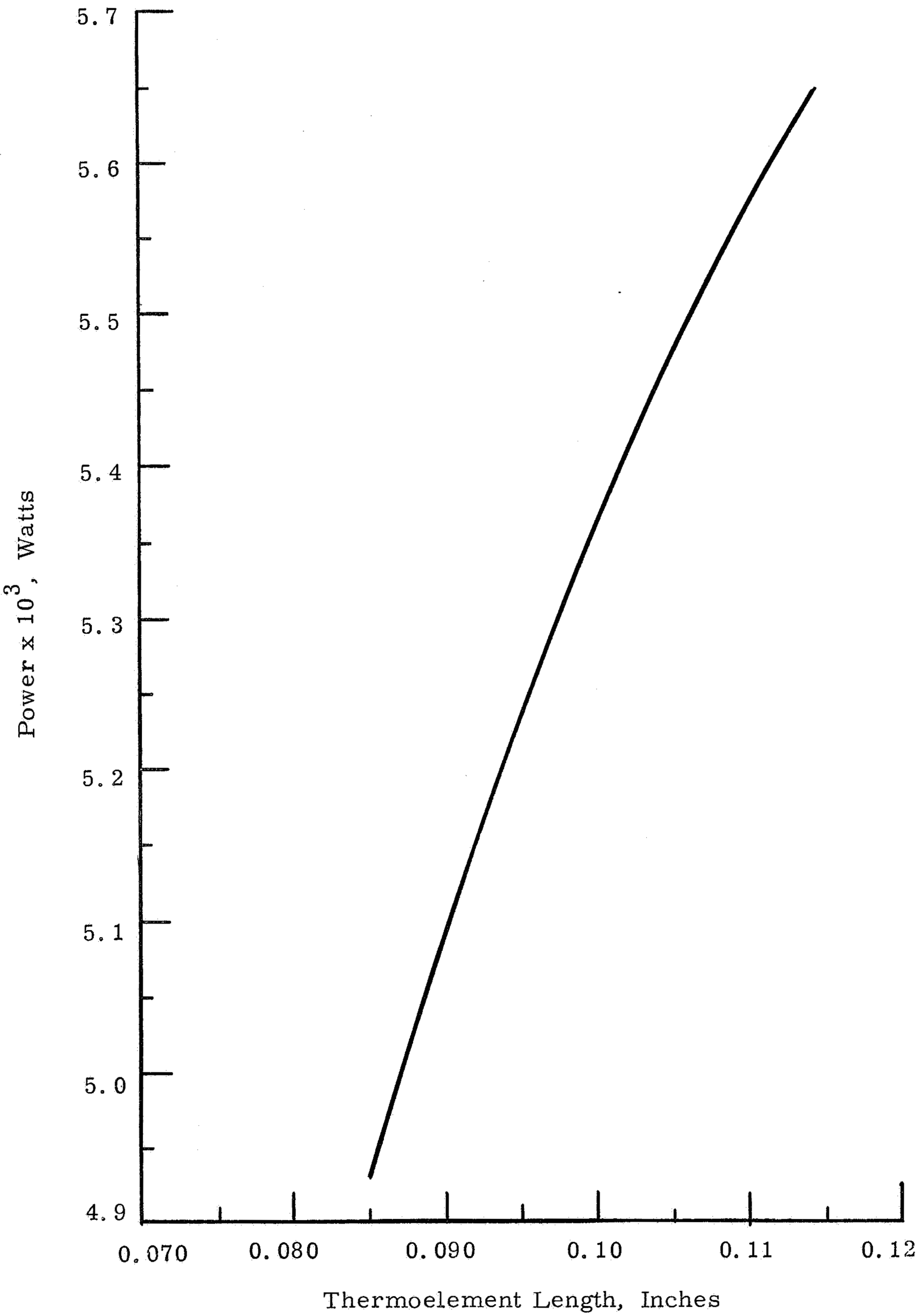


Figure IV-14. Thermoelement Length Parametric With Bi-Te Variable Properties

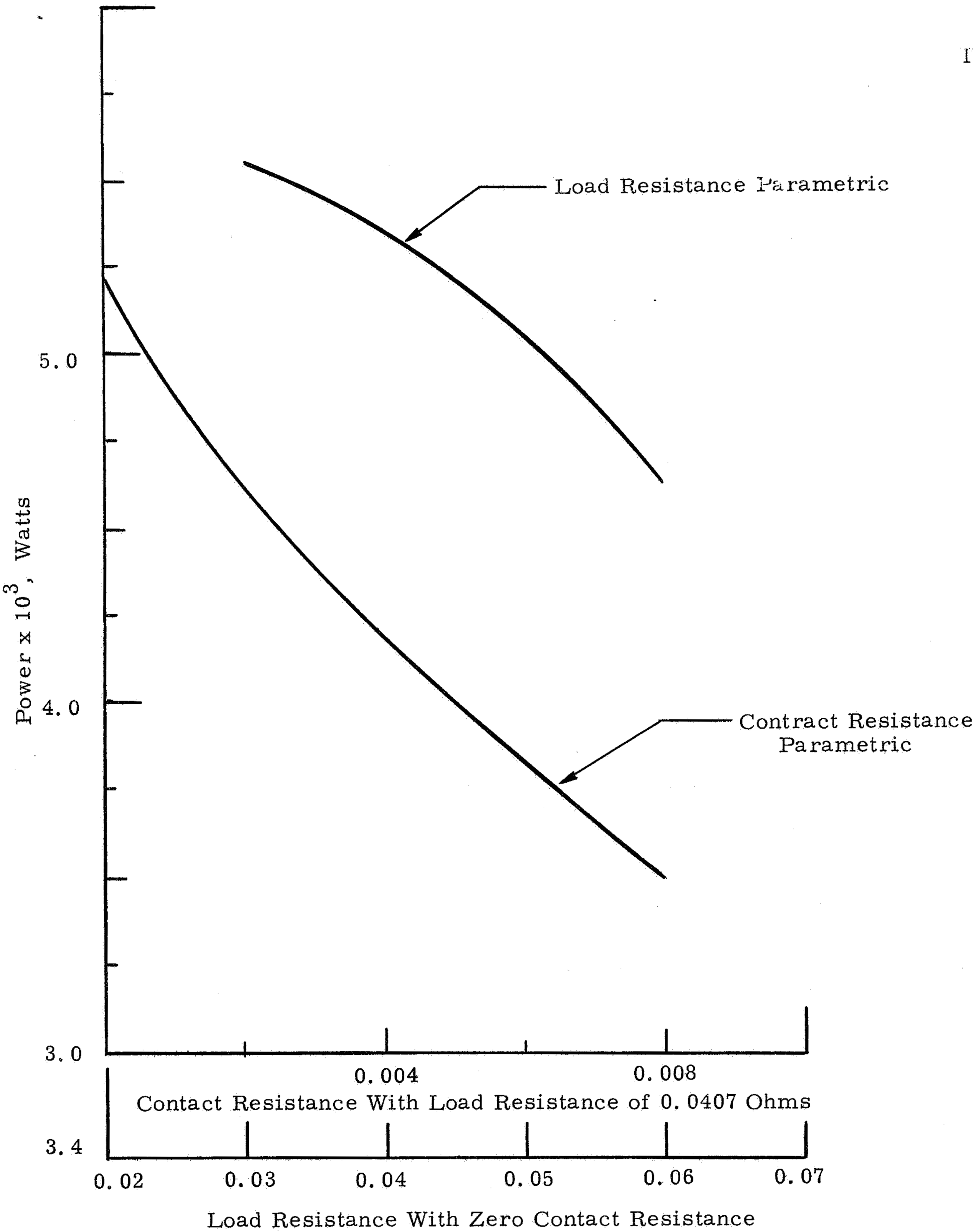


Figure IV-15. Bi-Te Variable Properties Resistance Parametric

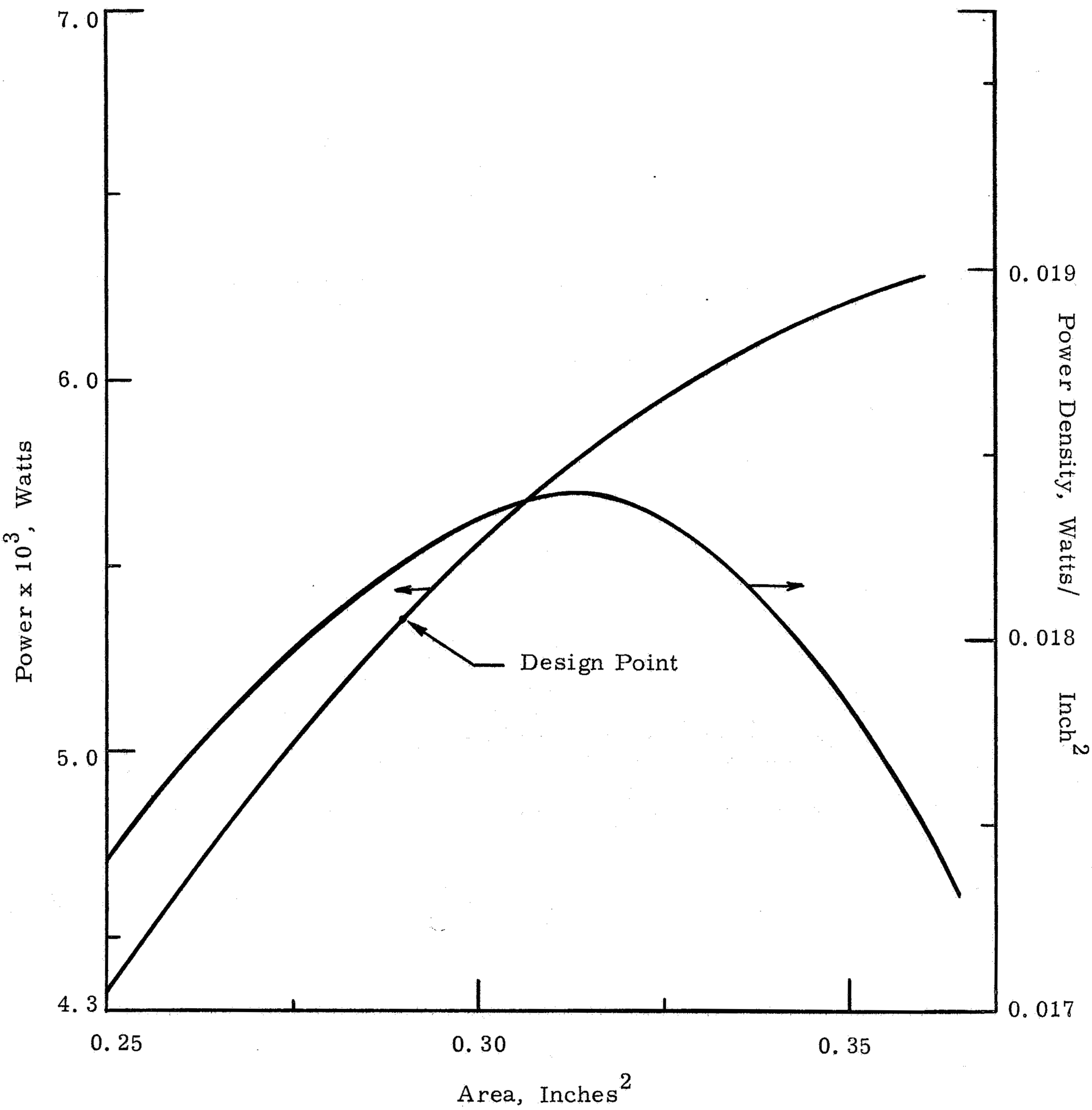


Figure IV-16. Plate Area Parametric Bi-Te Variable Properties

RCA for purposes of the calculation. These data are presented in Figures IV-17 through IV-19 (inclusive). The collector and radiator surface characteristics with the exception of the internal surface are shown in Figure IV-20.

TABLE IV-4. SOLAR FLAT PLATE CHARACTERISTICS FOR HIGH TEMPERATURE APPLICATIONS

Case I: Distance from Sun = 0.388AU	
Collector Material	Molybdenum
Radiator Material	Molybdenum
Thermoelement Material	Silicon-Germanium
Collector Thickness	0.005 in
Radiator Thickness	0.005 in
Thermoelement Length	0.24 in
Collector or Radiator Side Dimension	0.804 in
Thermoelement Side Dimension (equivalent)	0.088 in
Interplate Relative Emissivity (assumed)	0.03
Solar Flux	6w/in^2
*Case II: Solar Distance = 0.25AU	
Plate Side Dimension	0.574 in
Thermoelement Side Dimension (equivalent)	0.088 in
Solar Flux	14.4w/in^2

*NOTE: All other items identical to Case I values

A comparison of the characteristics computed by Bifano and those obtained by the computer program is shown in Table IV-5 for the Case I calculations.

TABLE IV-5. CALCULATED RESULTS FOR SOLAR FLAT PLATE AT 0.388AU FROM THE SUN

Item	Bifano	Computer Program
Hot Junction Temperature (^oR)	1660	1544.5
Cold Junction Temperature (^oR)	1000	969.40
Collector Temperature (^oR)	1660	1581.9

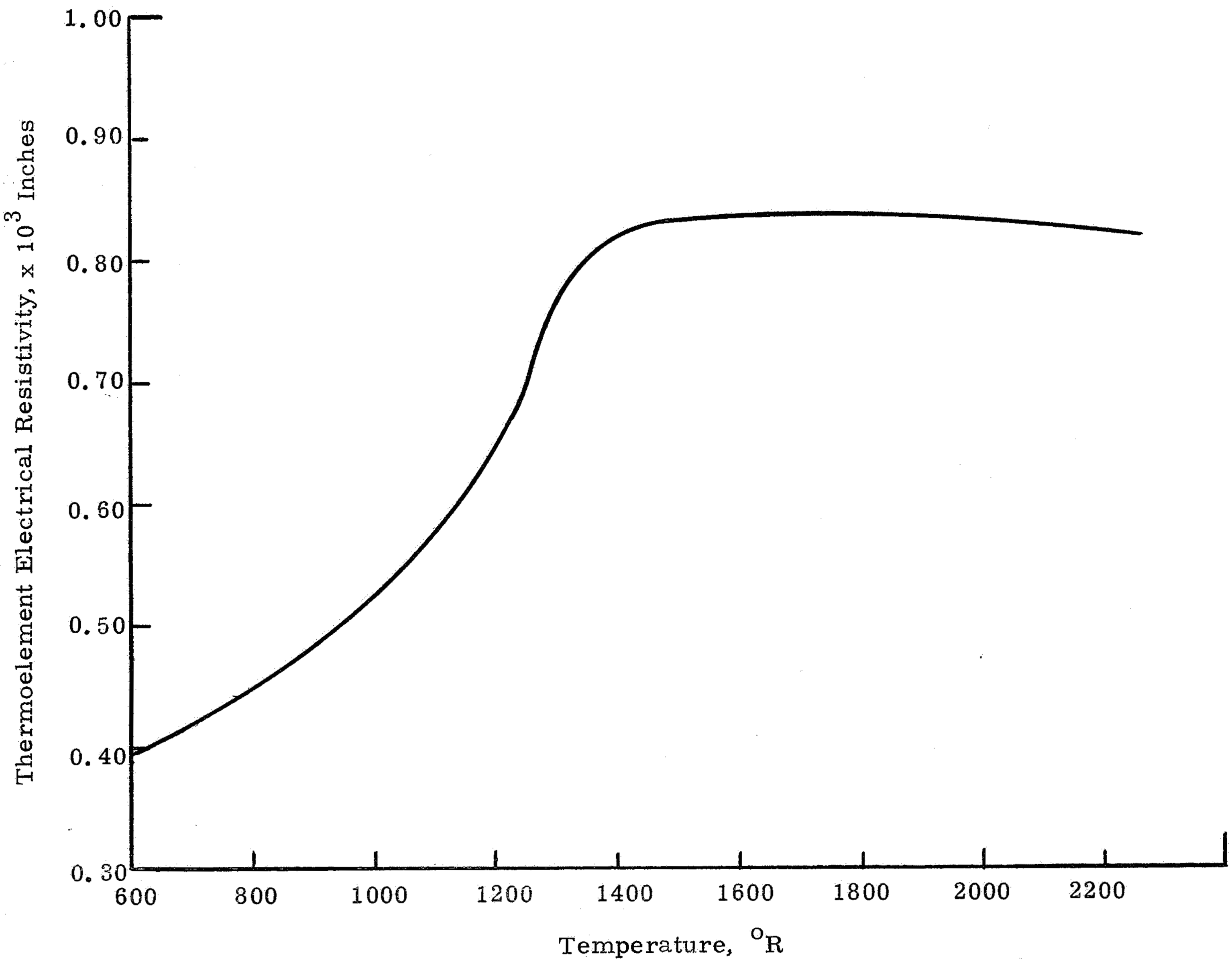


Figure IV-17. SiGe Electrical Resistivity

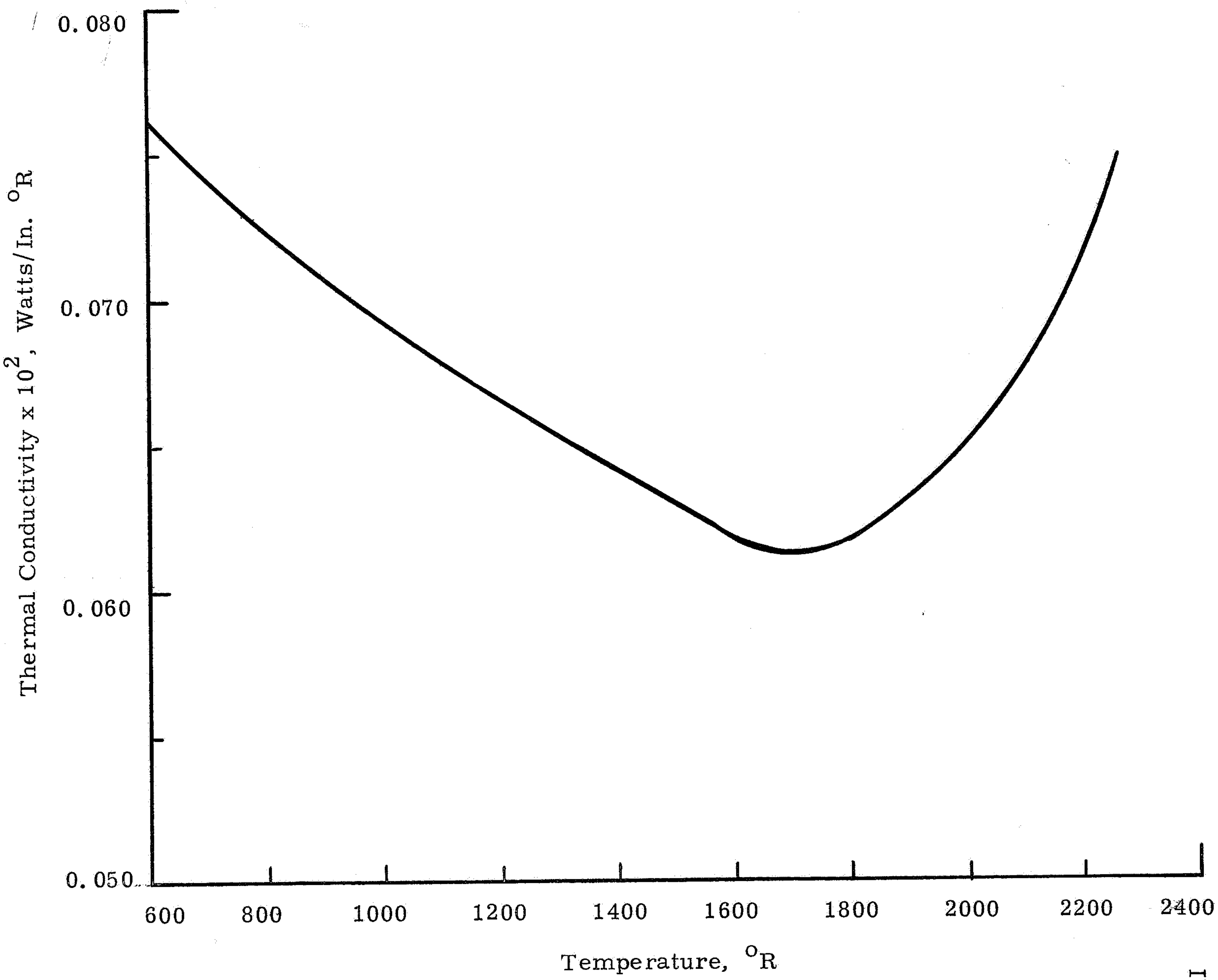


Figure IV-18, SiGe Thermal Conductivity

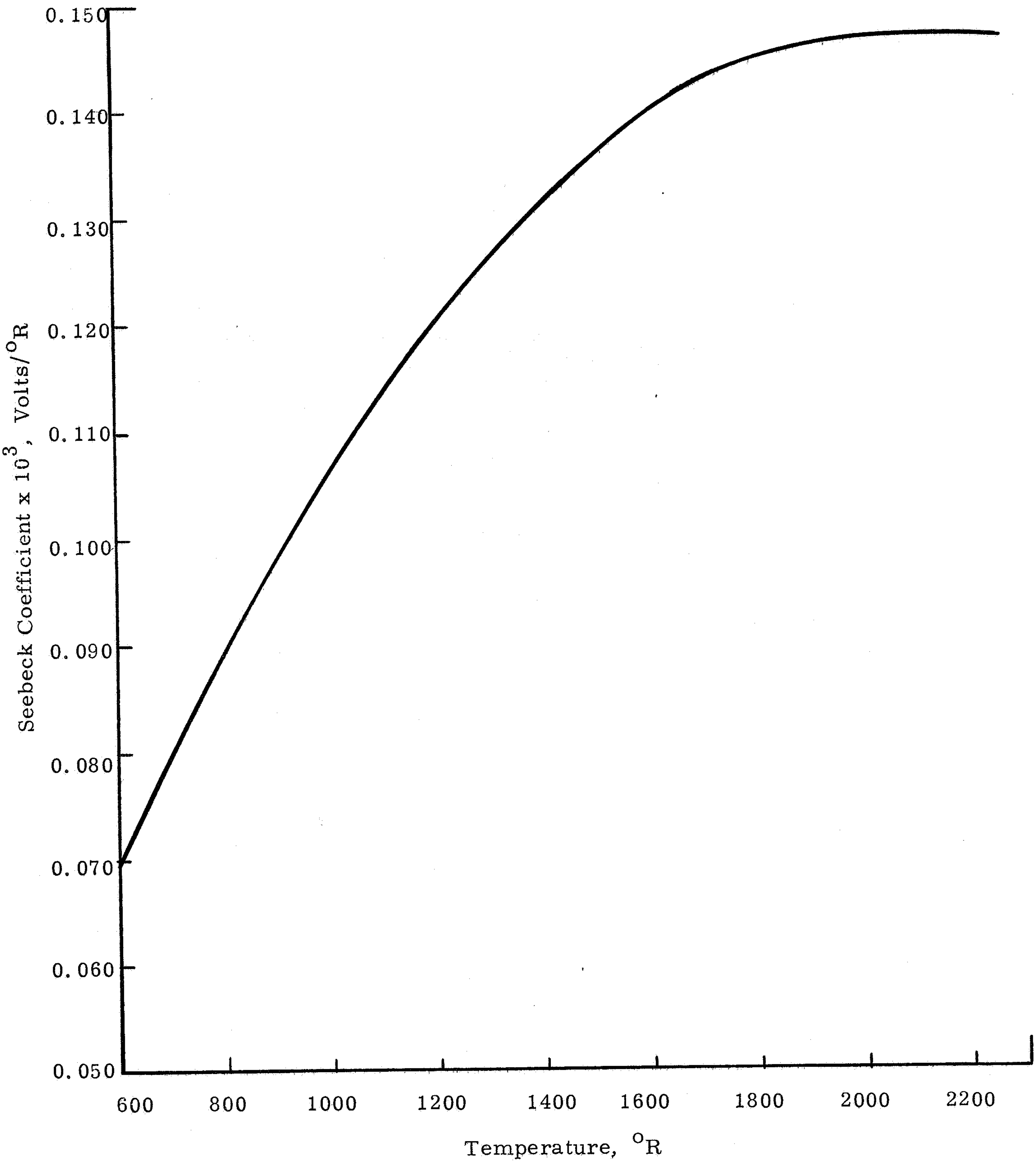


Figure IV-19. SiGe Seebeck Coefficient

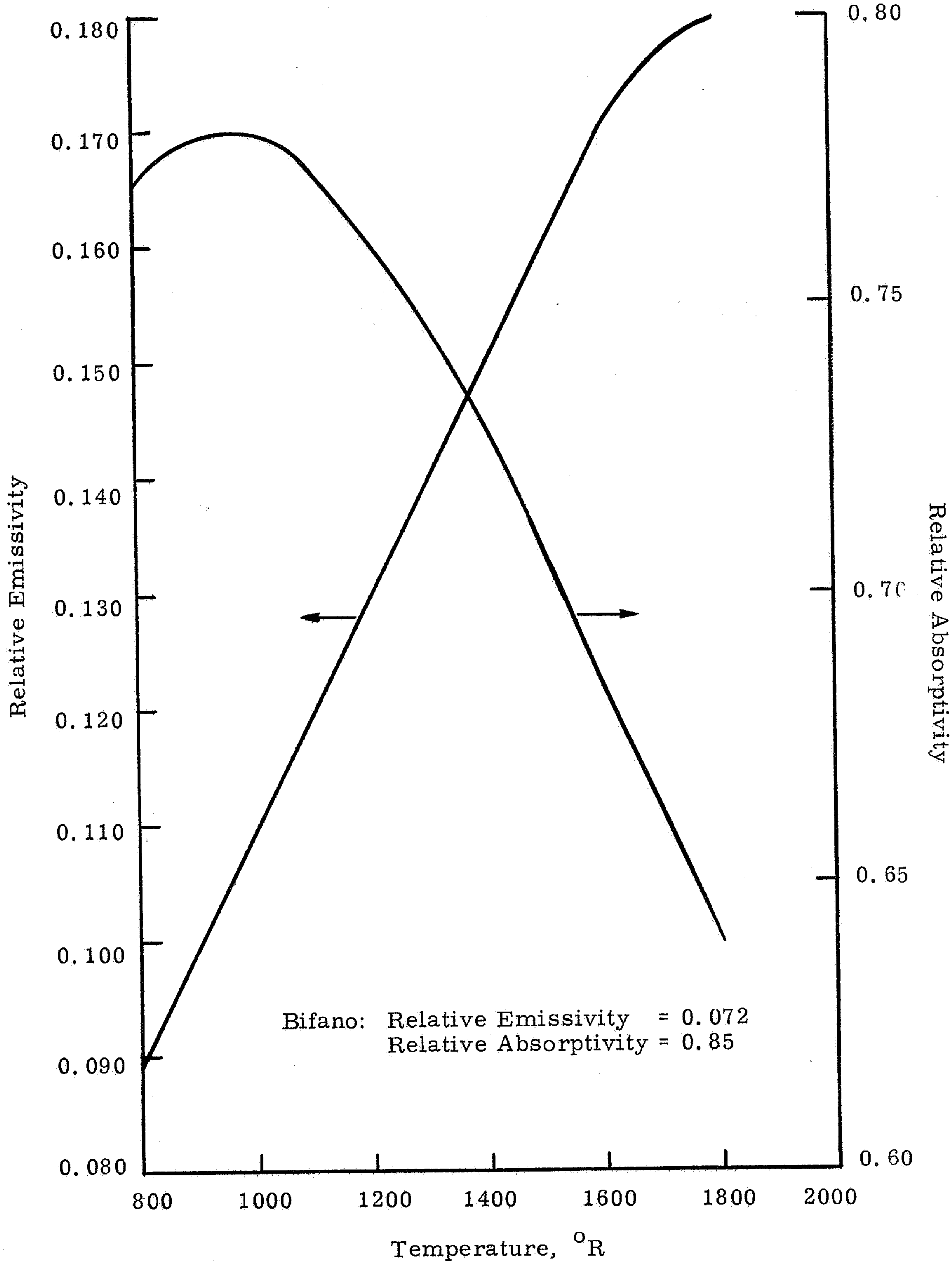


Figure IV-20. Absorber Surface Characteristics

TABLE IV-5. SOLAR FLAT PLATE CHARACTERISTICS FOR HIGH TEMPERATURE APPLICATIONS (CONTINUED)

Item	Bifano	Computer Program
Radiator Temperature (^oR)	1000	935.9
Power Density Based Upon Plate Area (watts/ft^2)	17	11.9

The difference between the calculated power output is due to the decreased temperature difference across the thermoelement which was predicted by the computer program. The reason for the sizeable drop in the collector temperature predicted by the computer program as compared to Bifano's calculations is due to the relative absorptivity of 0.85 which he used compared to the lower values used in the code. A secondary drop in the power output occurs because of the temperature difference in the collector and radiator from the center to the edges. Both of these effects act to decrease the temperature difference across the thermoelement which in turn decreases the output power.

No attempt was made during this investigation to optimize the thermoelement shape with respect to the area to length ratio or the thermoelement dimensions with respect to the plate area. However, the load resistance was varied to assure that the maximum power was being generated. The data presented in Table IV-5 were based upon the best load resistance.

The Case II calculations are summarized in Table IV-6. Again, most of the reason for the difference between the computer predicted power output and that shown by Bifano is due to the lower temperature difference across the thermoelement. This principally results from the lower collector temperature predicted by the computer, but secondarily results from the temperature gradient across the plate which Bifano's calculation could not investigate. Again, the computer results are based upon an optimized power output with respect to variation of the load resistance.

TABLE IV-6. CALCULATED RESULTS FOR SOLAR FLAT PLATE AT 0.25AU FROM THE SUN

Item	Bifano	Computer Program
Hot Junction Temperature (^oR)	2060	1909.5
Cold Junction Temperature (^oR)	1245	1208.3
Average Collector Temperature (^oR)	2060	1947.7
Average Radiator Temperature (^oR)	1245	1174.1
Power Density Based on Plate Area (watts/sq. ft.)	49.5	38.1

V. CONCLUSIONS

An accurate mathematical model of the solar flat plate has been obtained and a computer program written for the IBM-7094 has been utilized to obtain solutions for the mathematical model. The predicted characteristics agree closely with the experimental data which are available. Good correlation between the computer calculations and the previously available manual calculations have been obtained when suitable restrictions were placed upon the model utilized by the computer so that it corresponded to that upon which the manual calculations were based. Full use of the computer code capabilities shows a slight decrease in performance from that predicted for the existing configurations with manual calculations.

A number of calculations are possible with this computer program that previously could not be made. Transient behavior of the solar flat plate which results from movement from shade to sun or vice versa conditions may be predicted. The effect of variations in the absorber and radiator thickness can be determined so that true optimization with respect to weight can be considered.

Computer program input is very simplified and a comprehensive logical analysis of the input data is built into the code. Input data are read independent of format or the manner in which the data are written. Comprehensive parametric studies may be easily obtained because only those data which are changed from a previous run need be read by the computer program. Sufficient storage capability for tabular information has been incorporated into the computer program so that physical property data for several materials of interest may be stored and referred to by the computer as desired.

Although the program in its present form is quite generalized and applicable to most situations, its application to segmented or multiple thermoelements or the automatic calculation of orbital characteristics is not at present possible. These applications have been anticipated in developing the program and sufficient flexibility has been incorporated into the coding techniques so that the modifications may be made in a straightforward manner,if desired, at a later date.

APPENDIX A

DISCUSSION OF THE SELECTION OF A COORDINATE SYSTEM

APPENDIX A

Discussion of the Selection of a Coordinate System

Two basic approaches may be utilized in the solution of a heat conduction problem using numerical techniques. In the first, a differential equation represents the starting point, and the differential equation is expressed as a difference equation. This is solved by using suitable calculation techniques. In the second, the geometry which is to be analyzed is broken up into a number of small volumes, and each volume is considered to be represented by the properties of a point assigned within the volume. Normally, the points are called nodes and are considered as being interconnected to other nodes by conductances. The transient behavior of the volumes is represented by the "capacitance" of the volume, which may be obtained immediately from the heat capacity, the volume and the density. Representation in this technique essentially analyzes the equivalent electrical circuit to the heat conduction problem. Either approach should yield identical results if the analysis is applied properly.

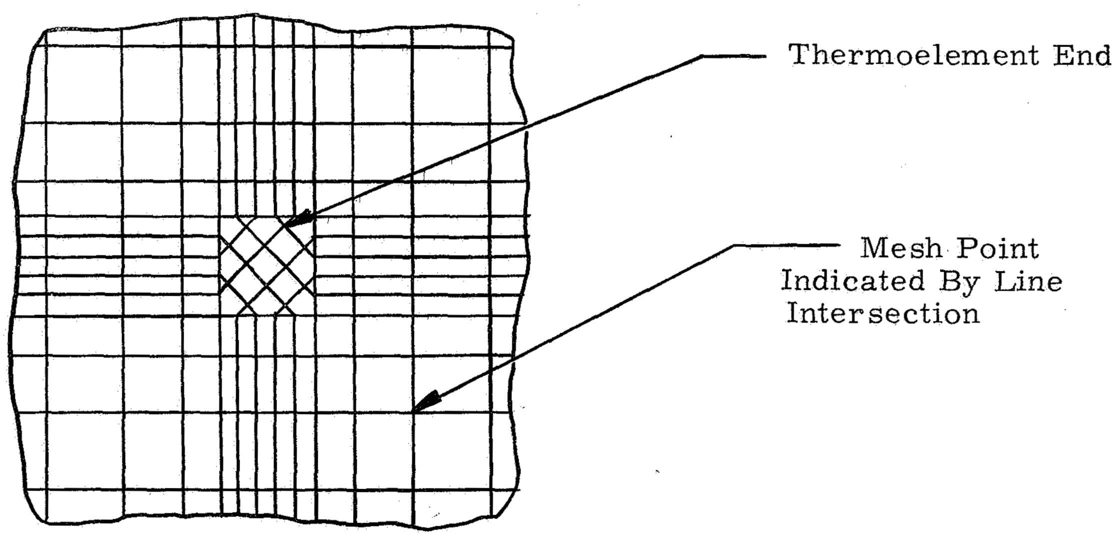
The differential equation applicable to the solar flat plate absorber or radiator is second order with respect to x and y in the Cartesian coordinate system. This means that if it is reduced to a difference equation, the temperatures in the steady state representation for a central node and four surrounding nodes will appear if a Taylor series expansion of the second derivative is used and higher order terms are neglected. It is common practice to solve equations of this type using equal spacing between all node centers. Since the absorber or radiator surface area is large compared to the cross sectional area of a thermoelement, a node size that was uniform everywhere would be determined by the minimum node size acceptable along a line of intersection between the thermoelement and the absorber or radiator, because the heat flux is highest at this line. Such an approach would result in a very large number of nodes being required for the analysis of the heat conduction. Although there is nothing wrong with this insofar as accuracy is concerned, the computation time increases very rapidly as the number of nodes is increased because the number of iterations required to attain suitable convergence becomes very large.

All of the heat absorbed at the hot end of the thermoelement must be conducted to that position from the absorber plate. Since the thermoelement end is small, the assumption was made that the thermoelement temperature at the hot junction could be represented by a single value. It is therefore reasonable to assume that the entire portion of the absorber formed by an extension of the thermoelement peripheral surfaces perpendicular to the plane of the absorber is also a uniform temperature region. This means that all of the heat collected by the absorber, with the small exception of that which is radiated directly to the small uniform region, must be conducted through this "thermoelement peripheral area". The heat fluxes in this region will be quite high, as will the temperature gradients. If an imaginary path is selected which is perpendicular to the periphery of the thermoelement and lies within the absorber, and one investigates temperature gradient while traveling along this path away from the thermoelement, it will be found that the gradient lessens and approaches zero at the edge of the absorber. Since the gradients are smaller further away from the thermoelement, larger sections of the absorber could be selected to be represented by a single node and an acceptable accuracy would still be attained. This immediately suggests that a numerical technique is desired wherein the size of the section assigned to a node is increased with increasing distance from the thermoelement. This will considerably reduce the required computation time with little or no loss in accuracy. An approach of this type is perhaps most easily understood by considering the electrical analog to heat transfer which was previously mentioned.

In utilizing this approach, it is vitally important to represent each node in a suitably designed configuration. It must be remembered that even though it may appear that a somewhat arbitrary approach is being taken, it is still necessary to mathematically represent the behavior. A haphazard sort of arrangement of interconnecting the nodes will not reduce to the necessary differential equation when the node size is allowed to approach zero. If the differential equation is not realized, then the required numerical solution may also not be realized.

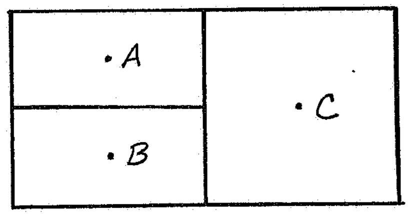
We therefore restrict the discussion for the present time to the selection of rectangular nodes which fit within the rectangular geometry which describes the absorber and connected thermoelement.

If we first consider that a large number of nodes must intersect the thermoelement-absorber intersection, and we wish to reduce the number of nodes as the distance from the thermoelement is increased, an arrangement of the following type can occur:

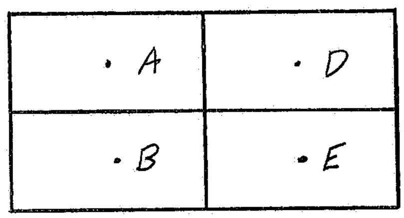


Although this reduces the number of points required for calculation purposes, we notice there are still a large number of extra points not required for accuracy. The ability to eliminate these is also important.

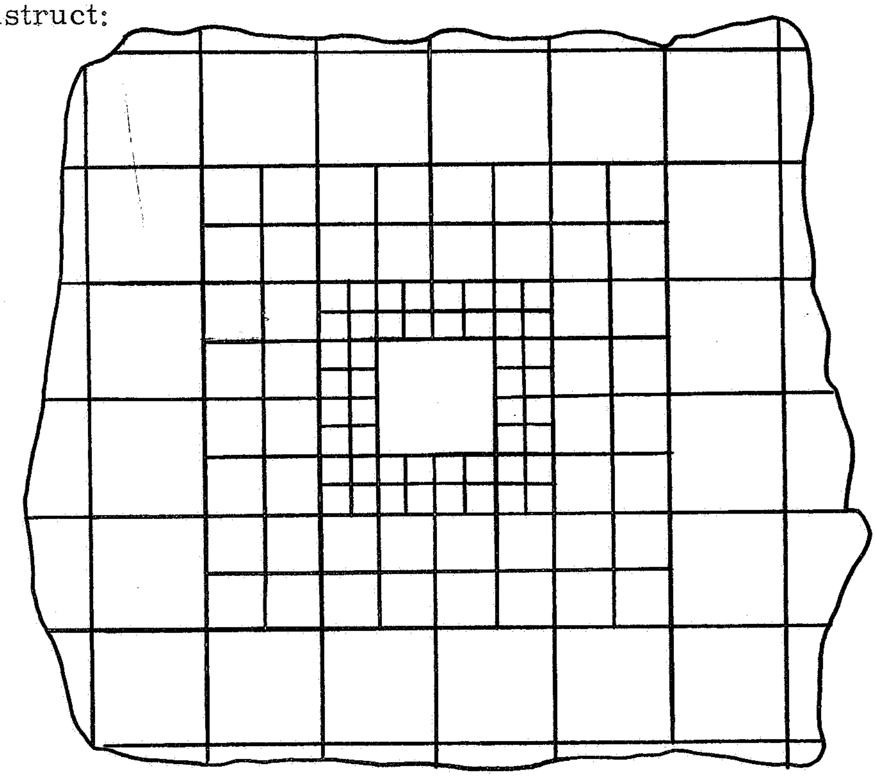
We now consider the following mesh point arrangement and surrounding area:



If the points A, B and C are properly interconnected so that this mesh is the equivalent of:



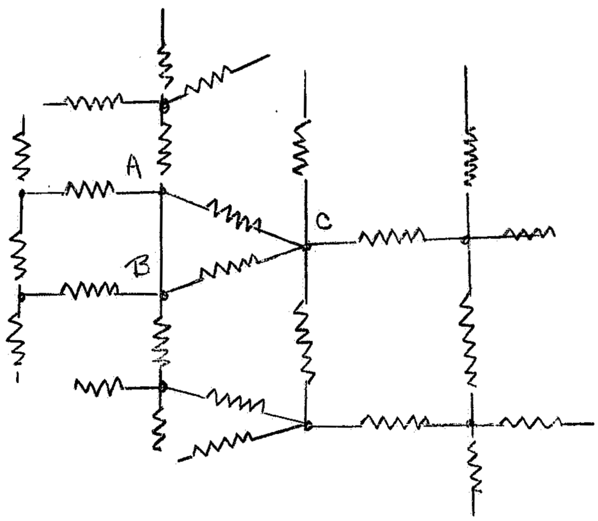
then we can construct:



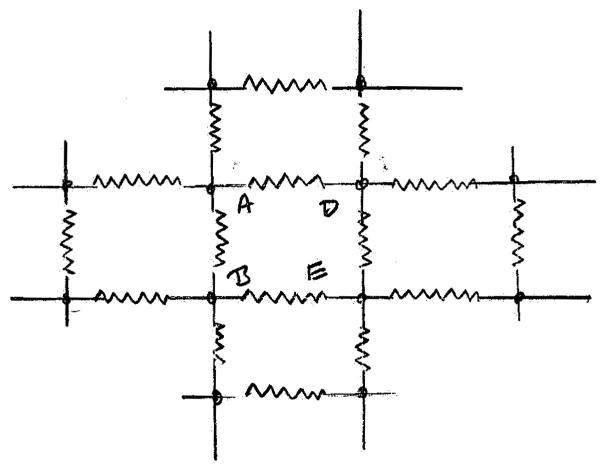
Here we see that the number of points is high near the thermoelement where the heat fluxes are high, and becomes lower as one moves away from the high heat flux area. The accuracy is maintained, and the number of mesh points reduced further.

To introduce "no" error in the system means that the conductances (or resistances) of each of the grids must give identical behavior of the mesh. Further, the capacitances associated with each of the points must also give the "same" overall mesh behavior. In our case, the grid is further complicated by the addition and loss of heat over the surface, which must also be included. Immediately, it is seen that the connecting of the mesh points is not just an arbitrary selection, but must be done carefully with a complete understanding of the heat conduction problem.

For this system to work properly, the electrical circuits shown in the following sketch must be identical:



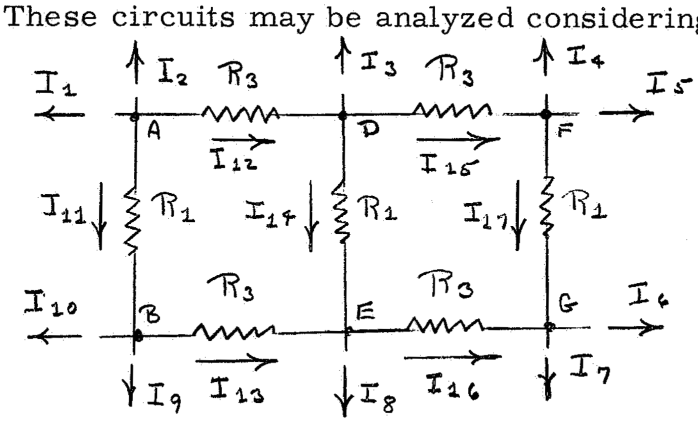
Reduction of Points Mesh



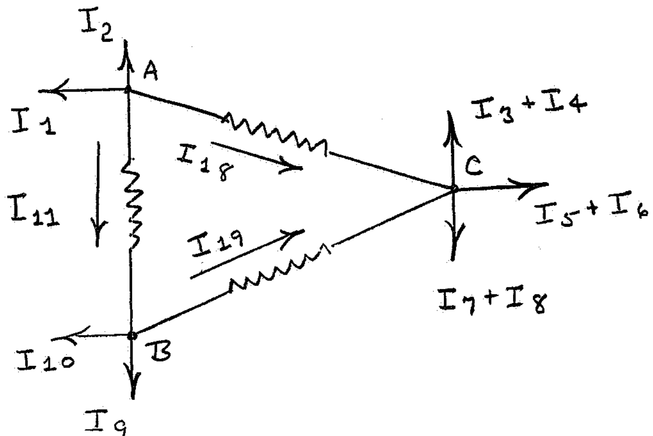
"Exact" Mesh

This requires careful calculation of all of the individual connectors, capacitances, etc., a problem generally ignored.

These circuits may be analyzed considering the following configurations:

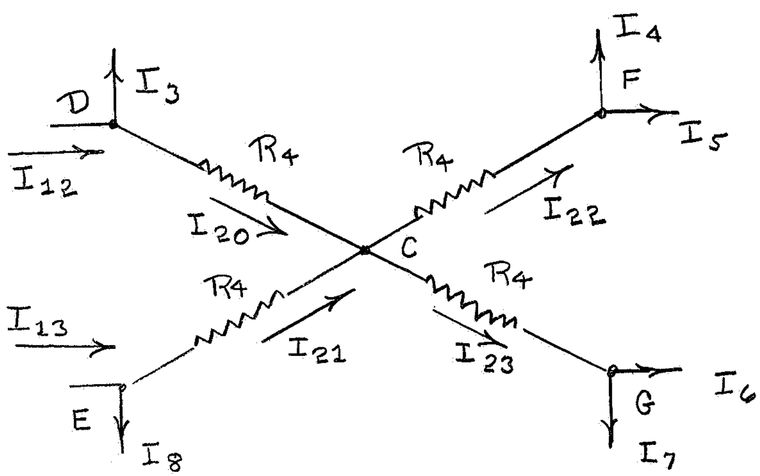


Circuit 1



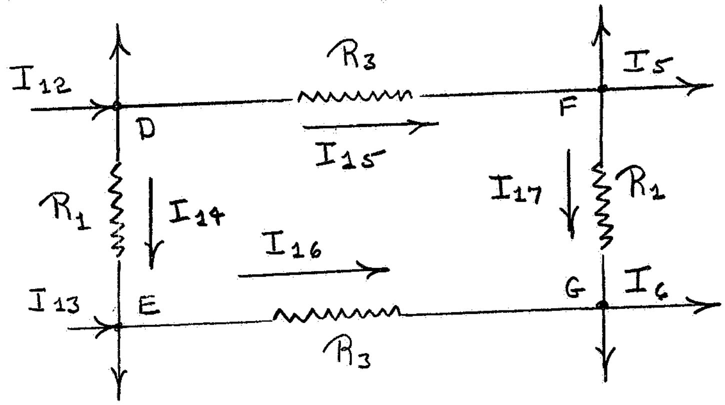
Circuit 2

With the additional stipulation that the point C be chosen such that:



Circuit 3

Where I = current, R = resistance, the first direction of which is indicated by an arrow. We first compare circuit 3 to:



Circuit 4

Immediately we see that:

$$V_D - I_{15} R_3 = V_F \tag{A-1}$$

$$V_D - I_{20} R_4 - I_{22} R_4 = V_F \tag{A-2}$$

Subtracting:

$$-I_{15} R_3 + \left[I_{20} + I_{22}\right] R_4 = 0 \tag{A-3}$$

Similarly:

$$I_{14} R_1 + \left[I_{21} - I_{20}\right] R_4 = 0 \tag{A-4}$$

$$I_{16} R_3 - \left[I_{23} + I_{21}\right] R_4 = 0 \tag{A-5}$$

$$-I_{17} R_1 + \left[-I_{22} + I_{23}\right] R_4 = 0 \tag{A-6}$$

Further:

$$I_{20} = I_{14} + I_{15} \tag{A-7}$$

$$I_{22} = I_{15} - I_{17} \tag{A-8}$$

$$I_{21} = -I_{14} + I_{16} \tag{A-9}$$

$$I_{23} = I_{16} + I_{17} \tag{A-10}$$

Using this, Equations (A-3) through (A-6) become:

$$-I_{15} R_3 + \left[I_{14} + I_{15} + I_{15} - I_{17}\right] R_4 = 0 \tag{A-11}$$

$$I_{14} R_1 + \left[-I_{14} + I_{16} - I_{14} - I_{15}\right] R_4 = 0 \tag{A-12}$$

$$I_{16} R_3 - \left[I_{16} + I_{17} - I_{14} + I_{16}\right] R_4 = 0 \tag{A-13}$$

$$-I_{17} R_1 + \left[-I_{15} + I_{17} + I_{16} + I_{17}\right] R_4 = 0 \quad \text{(A-14)}$$

which can be rewritten:

$$I_{15} \left[2R_4 - R_3\right] + I_{14} R_4 - I_{17} R_4 = 0 \quad \text{(A-15)}$$

$$I_{14} \left[-2R_4 + R_1\right] - I_{15} R_4 + I_{16} R_4 = 0 \quad \text{(A-16)}$$

$$I_{16} \left[-2R_4 + R_3\right] + I_{14} R_4 - I_{17} R_4 = 0 \quad \text{(A-17)}$$

$$I_{17} \left[2R_4 - R_1\right] - I_{15} R_4 + I_{16} R_4 = 0 \quad \text{(A-18)}$$

Now subtract Equation (A-17) from (A-15):

$$\left[I_{15} + I_{16}\right]\left[2R_4 - R_3\right] = 0 \quad \text{(A-19)}$$

Immediately:

$$R_4 = \frac{R_3}{2} \quad \text{(A-20)}$$

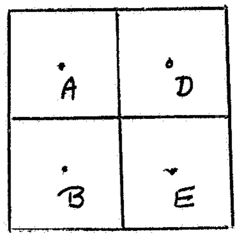
Finally, subtract Equation (A-18) from (A-16):

$$\left[I_{14} + I_{17}\right]\left[-2R_4 + R_1\right] = 0 \quad \text{(A-21}$$

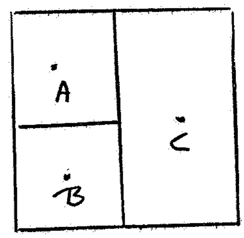
$$R_4 = \frac{R_1}{2} \quad \text{(A-22)}$$

But Equations (A-20) and (A-22) contradict unless $R_1 = R_3$, which immediately indicates that circuits 3 and 4 cannot be made equivalent except for square arrays, a prohibitive requirement since a non-square rectangular mesh is necessary for the general case. It is concluded that a mesh of the type indicated in configuration 2 cannot be converted to configuration 1 with the connecting point (C) in the center of the block (the circuit 3 requirement).

We therefore reject the requirement that we expand in both directions at once, and instead consider the change from configuration 2 to 3:

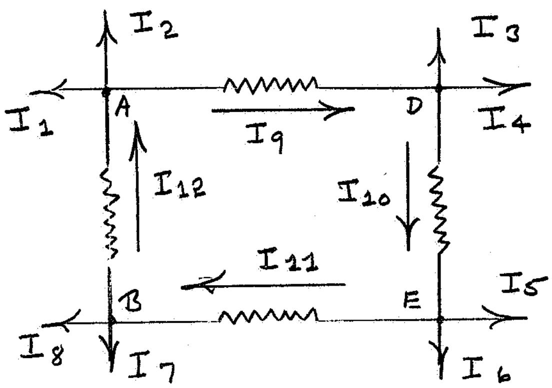


Configuration 2

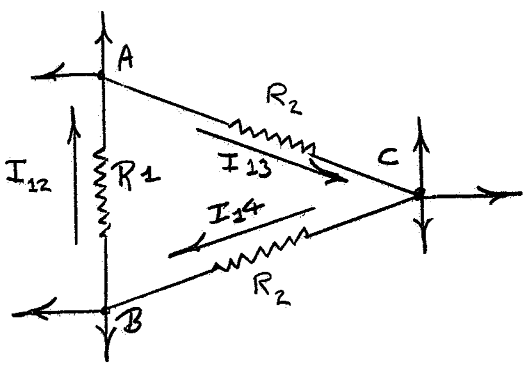


Configuration 3

which have the equivalent circuits:

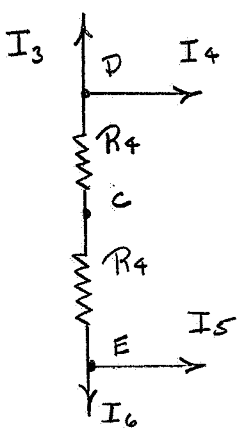


Circuit 5



Circuit 6

Here we require that:

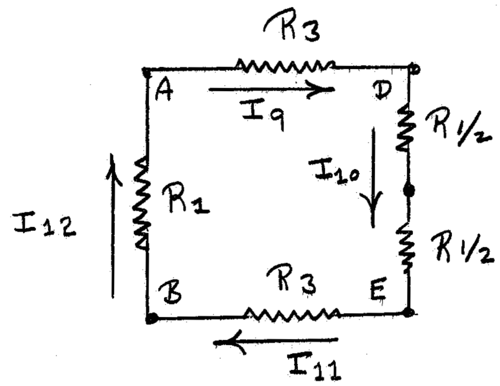


Circuit 7

since C is at the midpoint of D and E, and further that:

$$R_4 = \frac{R_1}{2} \qquad \text{(A-23)}$$

which fixes the position of C as in the midpoint of rectangle C. We immediately see that circuits 5 and 7 can be combined to:



Circuit 8

and this gives behavior identical to circuit 5. We now require that circuits 6 and 8 give identical behavior insofar as points A and B are concerned. Hence:

$$V_A - I_{13} R_2 = V_C \qquad \text{(A-24)}$$

$$V_C - I_{14} R_2 = V_B \qquad \text{(A-25)}$$

$$I_{13} = I_9 \qquad \text{(A-26)}$$

$$I_{14} = I_{11} \qquad \text{(A-27)}$$

$$V_A - I_9 R_3 = V_D \qquad \text{(A-28)}$$

$$V_D - I_{10} \frac{R_1}{2} = V_C \qquad \text{(A-29)}$$

$$V_C - I_{10} \frac{R_1}{2} = V_E \qquad \text{(A-30)}$$

$$V_E - I_{11} R_3 = V_B \tag{A-31}$$

Now subtract Equation (A-30) from (A-29):

$$V_D - \cancel{I_{10} \frac{R_1}{2}} - V_C + \cancel{I_{10} \frac{R_1}{2}} = V_C - V_E \tag{A-32}$$

$$2V_C = V_D + V_E \tag{A-33}$$

$$V_C = \frac{V_D + V_E}{2} \tag{A-34}$$

(which would be expected).

Now substitute Equations (A-28) and (A-31) in (A-34):

$$V_C = \frac{V_A - I_9 R_3 + V_B + I_{11} R_3}{2} \tag{A-35}$$

Now we use Equations (A-24) through (A-27) with (A-35) to obtain:

$$2V_C = V_C + I_9 R_2 - I_9 R_3 + V_C - I_{11} R_2 + I_{11} R_3 \tag{A-36}$$

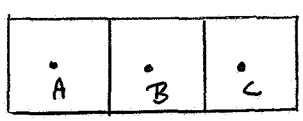
$$I_9 \left[R_2 - R_3\right] - I_{11} \left[R_2 - R_3\right] = 0 \tag{A-37}$$

$$\left[I_9 - I_{11}\right] \left[R_2 - R_3\right] = 0 \tag{A-38}$$

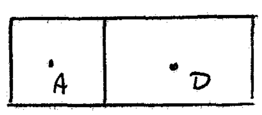
which for the most general behavior requires simply that:

$$R_2 = R_3 \tag{A-39}$$

This enables us to expand the mesh in one direction (say, for example, the y direction). We now must derive a technique that will enable expansion in the other direction (the x direction). Hence we require that the following configurations be equivalent:

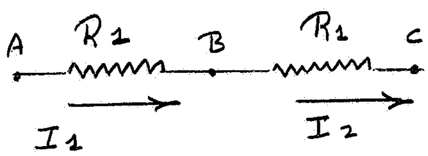


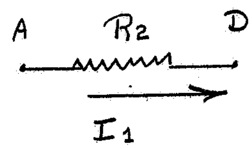
Configuration 4



Configuration 5

The equivalent circuits are:





Circuit 9 Circuit 10

Since we are doubling size, and D is to be placed at the midpoint, there results:

$$R_2 = \frac{3R_1}{2} \tag{A-40}$$

as the requirement that positions point D. Further, we see that:

$$V_A - I_1 R_1 = V_B \tag{A-41}$$

$$V_A - I_1 R_2 = V_D \tag{A-42}$$

which combine to form:

$$V_A - I_1 \frac{3R_1}{2} = V_D \tag{A-43}$$

$$\frac{2V_A}{3} - I_1 R_1 = \frac{2V_D}{3} \tag{A-44}$$

$$\frac{V_A}{3} = V_B - \frac{2V_D}{3} \tag{A-45}$$

$$V_A = 3V_B - 2V_D \tag{A-46}$$

$$V_D = \frac{3V_B - V_A}{2} \tag{A-47}$$

which is reasonable weighting and replacing circuit 9 by 10 is therefore reasonable. Similar changes may be made if other than doubling of the mesh size is desired.

This academic type discussion indicates that the node interconnection problem is more complicated than generally realized. It further indicates that if the rectangular type of coordinate system is to be utilized, it is going to be difficult to obtain suitable expansion factors in going from one node size to another so that suitably sized nodes can be fitted to the different requirements of the plate edge as compared to the intersection of the plate with the thermoelement.

It is not absolutely necessary that the electrical analogy hold identically, as implied in the foregoing analysis. If suitable resistances are selected which come close to representing the overall behavior, suitable answers will still be attained. Such a relaxation of the rigorousness of the approach is commonly utilized, but the authors suspect that the assumptions involved are not fully understood.

One of the basic requirements imposed upon the analysis was that the resulting computer program should be "easy to use". The authors concluded early in the study of mesh arrangements that the only information necessary for general prescription of the solar flat plate was the coordinates of the corners of the plate and of the thermoelement along with specifications in regard to the number of mesh points which should be used. A mesh generation technique which was easier to set up in a logical manner than the rectangular approach just discussed was therefore desired. As a result, the rectangular system was rejected in favor of the cylindrical coordinate system.

The cylindrical coordinate system with the origin centered at the thermoelement fits the requirements for an expanding mesh very nicely. The width of "rays" close to the thermoelement is small, and increases with increasing distance from the thermoelement. Selection of increment sizes such that an approximately "square" node results allows the automatic generation of the desired nodal arrangement.

A complication with the cylindrical geometry approach that was not evident in the rectangular geometry approach is the "non-fit" of the geometry to the thermoelement-plate intersection and to the outside edges of the plate. The logic involved in these sections was easier to set up than was the logic involved in generating an expanding generalized node system based upon rectangular geometry. Cylindrical geometry was therefore selected as the basis for the calculations.

APPENDIX B

PROGRAM LISTING

```
$IBFTC MAIN     LIST,DECK,DD,XR7
C       F L A T   P L A T E   T / E   G E N E R A T O R   A N A L Y S I S
C
C   C. ANDERSON AND W. LYON, HITTMAN ASSOCIATES, BALTIMORE, MD., 1966.
C
      COMMON Z
      DIMENSION Z(20000),MA2(12),JS(6),MS(12),ITAB(10),ITABA(30),ICA(200
     2),ICS(200)
      EQUIVALENCE (Z(1),IN),(Z(2),IOUT),(Z(3763),MS(1)),(Z(614),MA2(1)),
     2(Z(827),JS(1)),(Z(13),LINE),(Z(12),IPR),(Z(2443),NP3),(Z(640),
     3NOLU),(Z(663),NP4),(Z(2433),ITAB(1)),(Z(15940),ITABA(1)),(Z(15740)
     4,ICA(1)),(Z(2233),ICS(1)),(Z(3746),JZZ),(Z(11),LERR),(Z(3),IQUIT),
     5(Z(6316 ),A(1)),(Z(9016),CENX(1)),(Z(9916),CENY(1)),
     6(Z(8116),PLEIN(1)),(Z(10816),NODE(1)),(Z(7216),PLEBAK(1)),
     7(Z(12816),CONDB(1)),(Z(11916),CONDI(1)),(Z(13716),T(1)),
     8(Z(7216),XK(1)),(Z(8116),CAP(1)),(Z(14616),EPSC1(1)),
     9(Z(15066),EPSC2(1)),(Z(15066),EPSR1(1)),(Z(15516),EPSR2(1))



      EQUIVALENCE (Z(9016),ALFM(1)),(Z(9916),T4(1)),(Z(16416),TERMF(1)
     1),(Z(10816),TERMO(1))    ,(Z(4409),TMAP),(Z(5200),LB,L22),
     3(Z(5213),L2),(Z(5214),ITOT,IX) ,
     4(Z(16416),NOD(1)),(Z(14616),XPRNT(1)),(Z(15516),YPRNT(1)),
     5(Z(4408),XINPUT)  ,(Z(4),K),(Z(4410),ISS)
      NAMELIST/NAM1/Z
C     SET VARIABLES FOR PROBLEM USING ALL NEW DATA                      MAI
  103 DO 260 I=1,20000
  260 Z(I)=0.
C     SET TAPE DESIGNATIONS, IN IS INPUT, IOUT OUTPUT                   MAI
      IN=5                                                              MAI
      IOUT=6                                                            MAI
C     SET VARIABLES FOR PROBLEM USING PARTIAL NEW DATA                  MAI
  239 DO 3 I=2,6
    3 JS(I)=0                                                           MAI
C     READ INFORMATION
    9 CALL READIT(JZZ)
      IF(IQUIT)103,210,210
C     PROCESS NUMBERS FOR CORRECT LOGIC AND PLACE IN CORRECT LOCATIONS.
C     JS IS * INFORMATION, CONTAINS 0 IF NOTHING WAS READ.
  210 IF(JS(3)+JS(4))100,4,26
   26 CALL PACK
      GO TO 4
  100 WRITE(IOUT,101)
  101 FORMAT(32H0MACHINE ERROR - I GOOFED - MAIN)
   11 CALL EXIT
    4 DO 10II=2,6
      GO TO (100,12,14,13,15,16),II
   12 IF(JS(2))100,10,6
    6 CALL PRO2
   18 IF(LERR)100,10,17
   17 JZZ=1
      IF(K-8)9,103,11
   13 IF(JS(5))100,10,29
   29 CALL PRO5
      GO TO 18
   14 IF(JS(6))100,10,30
   30 CALL PRO6
      GO TO 18
```

```
15 IF(JS(3))100,10,31
31 I=JS(4)
   JS(4)=0
   CALL UNPAK
C  SAVE LOWER MEMORY
   DO 35 J=1,1580
35 Z(J+10249)=Z(J+3758)
   CALL PRO3
C  RESTORE LOWER MEMORY
   DO 36 J=1,1580
36 Z(J+3758)=Z(J+10249)
   JS(4)=I
   GO TO 18
16 IF(JS(4))100,10,37
37 I=JS(3)
   JS(3)=0
   CALL UNPAK
   CALL PRO4
   JS(3)=I
   GO TO 18
10 CONTINUE
   IF(XINPUT)310, 10,309



310 CALL MESH
    IF(LERR)100,251,17
251 CALL MESHA(A,CENX,CENY,PLEBAK,PLEIN,NODE,IX,L2)
    IF(LERR)100,252,17
252 CALL MESHB(NODE,A,CENX,CENY,PLEBAK,PLEIN,IX,L2)
    CALL MESHC(CONDB,NODE,CONDI,PLEBAK,CENX,CENY,A,T,PLEIN,IX,L2)
309 IF(TMAP)201,202,201
201 CALL     FPMAP(CENX,CENY,NOD,XPRNT,YPRNT,IX,L2)
202 IF(ISS)305,305,304
304 CALL CALCS(XK,CAP,EPSC1,EPSC2,EPSR1,EPSR2,ALPH,T,A,T4,
   1CONDI,CONDB,TERMF,TERMO,IX,L2,L22)
    GO TO 306
305 CALL CALCT(XK,CAP,EPSC1,EPSC2,EP        ,ALPH,T,A,T4,
   1CONDI,CONDB,TERMF,TERMO,IX, 2,L
306 CONTINUE
    IF(IQUIT)100,238,11
238 IF(K-6)239,103,11
    END
```

```
$IBFTC READIT   LIST,DECK,DD,XR7
      SUBROUTINE READIT(JZZ)                                              READIT
      COMMON Z                                                            READIT
      DIMENSION Z(16000),ISS(72),IS(72)                                   READIT
      EQUIVALENCE (Z(1),IN),(Z(2),IOUT),(Z(3),IQUIT),(Z(4),K),(Z(5),NUM)  READIT
     2,(Z(6),IGN),(Z(7),DEC),(Z(8),NUMB),(Z(9),ZEXP),(Z(10),DECP),(Z(11)  READIT
     3,LERR),(Z(13),LINE),(Z(9900),ISS(1)),(Z(9800),IS(1)),(Z(832),MACH)  READIT
     4,(Z(3747),KB),(Z(7900),IG),(Z(7901),ITE),(Z(7902),TEMP)             READIT
      INTEGER A,AST,ZZ,BLANK,DOT,PLUS,E                                   READIT
      DATA A,AST,ZZ,BLANK,DOT/1HA,1H*,1HZ,1H ,1H./,PLUS/1H+/,E/1HE/       READIT
C CHECK FOR NORMAL ENTRY,321 IS YES                                       READIT
      LERR=0                                                              NUMBER
      IF(JZZ)100,321,32                                                   READIT
  321 K=0                                                                 READIT
   18 READ (IN,1) (IS (I),I=1,72)                                         READIT
    1 FORMAT(72A1)                                                        READIT
      WRITE (IOUT,500)(IS(I),I=1,72)
  500 FORMAT (1H0,72A1)
      IA=1                                                                READIT
C CHECK FOR *                                                             READIT
      IF(IS(1)-AST)6,3,6                                                  READIT
C CONTROL CARD, DETERMINE TYPE                                            READIT
    3 IS(2)=IS(2)/1073741824
      IF(IS(2))9,9,10                                                     READIT
    9 IS(2)=IS(2)*1073741824
      WRITE(IOUT,500)(IS(I),I=1,72)
      WRITE(IOUT,12)                                                      READIT
   12 FORMAT(21H0ILLEGAL CONTROL CARD)                                    READIT
  322 CALL EXIT                                                           READIT
   10 IF(9-IS(2))9,13,14                                                  READIT
   13 IQUIT=1                                                             READIT
      RETURN                                                              READIT
   14 IF(7-IS(2))16,16,17                                                 READIT
  100 WRITE(IOUT,101)MACH                                                 READIT
  101 FORMAT(42H0MACHINE ERROR IN READIT, STATEMENT NUMBER I4)            READIT
      CALL EXIT                                                           READIT
   16 IQUIT=0                                                             READIT
      RETURN                                                              READIT
   17 K=IS(2)                                                             READIT
      IA=3                                                                READIT
    6 IF(K-1)102,19,20                                                    READIT
  102 WRITE(IOUT,103)                                                     READIT
  103 FORMAT(21H0MISSING CONTROL CARD)                                    READIT
      CALL EXIT                                                           READIT
C COMMENT CARD WAS READ                                                   READIT
   19 IS(1)=BLANK                                                         READIT
      IS(2)=BLANK                                                         READIT
      WRITE(IOUT,500)(IS(I),I=1,72)
      LINE=LINE+1                                                         READIT
      GO TO 18                                                            READIT
C CALCULATE NUMERICAL DATA                                                READIT
   20 NUM=0                                                               READIT
      IGN=0                                                               READIT
      DEC=0.                                                              READIT
      NUMB=0                                                              READIT
      KB=0                                                                READIT
      ZEXP=0.                                                             READIT
      DECP=0.                                                             READIT
```

```
      JZZ=0
      LINE=0                                                            READIT
      IF(K-8)410,418,410
  410 IF(K-9)400,401,402                                                READIT
  402 MACH=32                                                           READIT
      GO TO 100                                                         READIT
  418 IQUIT=-1
      RETURN
  401 CALL EXIT                                                         READIT
C  SEARCH FOR NEW PROBLEM                                               READIT
  400 READ(IN,405)IS(1),IS(2)                                           READIT
      IF(IS(1)-AST)400,406,400                                          READIT
  405 FORMAT(2A1)                                                       READIT
  406 IS(2)=IS(2)/1073741824
      IF(IS(2)-8)408,418,408
  408 IF(IS(2)-9)400,401,400                                            READIT
      END                                                               READIT
```

```
$JOB    1202C003 427 NOCJA N0001B1270
$EXECUTE          IBJOB
$IBJOB            MAP,FIOCS
$IBFTC NUMBER   LIST,DECK,DD,XR7
      SUBROUTINE NUMBER(KB)                                             NUMBER
C                                                                       NUMBER
      COMMON Z                                                          NUMBER
      DIMENSION Z(16000),I2(200),N2(200),P2(200),I3(1600),N3(1600),P3(  NUMBER
     21600),I4(1830),N4(1830),P4(1830),N5(730),P5(730),I5(730),N6(600), NUMBER
     3P6(600),I6(600)                                                   NUMBER
      EQUIVALENCE (Z(2),IOUT),(Z(4),K),(Z(5),NUM),(Z(6),IGN),(Z(7),DEC),NUMBER
     2(Z(8),NUMB),(Z(9),ZEXP),(Z(10),DECP),(Z(11),LERR),(Z(13),LINE),(Z(  NUMBE
     33575),I2(1)),(Z(3375),N2(1),P2(1))          ,(Z(828),J2),(Z(829),J3),NUMBER
     4(Z(830),J4),(Z(831),J5),(Z(832),J6),(Z(5379),I3(1)),(Z(3779),N3(  NUMBER
     51)),(Z(3779),P3(1)),(Z(10000),I4(1)),(Z(11830),P4(1)),(Z(11830),  NUMBER
     6N4(1)),(Z(6991),N5(1)),(Z(6991),P5(1)),(Z(7721),I5(1)),(Z(8451),  NUMBER
     7N6(1)),(Z(8451),P6(1)),(Z(9051),I6(1)),(Z(9651),I),(Z(9652),TEMP),NUMBER
     8(Z(9653),TEM),(Z(9654),TEMA)                                      NUMBER
C     SET ERROR RETURN FOR NO ERROR                                     NUMBER
      LERR=0
      K=K                                                               NUMBER
C     CHECK FOR NUMBER OR Z                                             NUMBER
    7 IF(KB)100,3,6                                                     NUMBER
    6 I=-1                                                              NUMBER
      GO TO 5                                                           NUMBER
C     DETERMINE TYPE OF NUMBER                                          NUMBER
    3 IF(DEC)100,1,2                                                    NUMBER
  100 WRITE(IOUT,101)                                                   NUMBER
  101 FORMAT(24HOMACHINE ERROR IN NUMBER)                               NUMBER
      LERR=1                                                            NUMBER
      RETURN
    1 IF(IGN)2,4,2                                                      NUMBER
C     FIXED POINT NUMBER, SET SIGN                                      NUMBER
    4 NUMB=NUM*NUMB                                                     NUMBER
      I=0                                                               NUMBER
      GO TO 5                                                           NUMBER
C     FLOATING POINT NUMBER, COMPILE PIECES                             NUMBER
    2 TEMP=NUMB                                                         NUMBER
      TEM=NUM                                                           NUMBER
      TEMA=IGN                                                          NUMBER
      PNUMB=(TEMP+DECP)*TEM*10.**(TEMA*ZEXP)                            NUMBER
      I=1                                                               NUMBER
C     BRANCH TO TYPE OF INPUT AND STORE                                 NUMBER
    5 GO TO (100,12,13,14,15,16),K                                      NUMBER
   12 J2=J2+1                                                           NUMBER
   18 IF(201-J2)102,102,17                                              NUMBER
  102 WRITE(IOUT,103)K                                                  NUMBER
  103 FORMAT(2H0*I1,45HTABLE OVERFLOW - YOU JUST KAUNT USE THIS MUCH)   NUMBER
      LERR=1
   19 RETURN                                                            NUMBER
   17 I2(J2)=I                                                          NUMBER
      IF(I)19,20,21                                                     NUMBER
   20 N2(J2)=NUMB                                                       NUMBER
      GO TO 19                                                          NUMBER
   21 P2(J2)=PNUMB                                                      NUMBER
      GO TO 19                                                          NUMBER
   13 J3=J3+1                                                           NUMBER
   22 IF(1410-J3)102,102,23                                             NUMBER
```

```
   45 IF(DEC)46,47,147
   46 MACH=45                                                         READIT
      GO TO 100                                                       READIT
   47 NUMB=NUMB*10                                                    READIT
      GO TO 21                                                        READIT
  147 DEC=DEC*10.
      GO TO 21
 CHECK NEXT VALUE FOR BLANK                                           READIT
   44 IF(72-I)49,62,61                                                READIT
   49 MACH=44                                                         READIT
      GO TO 100                                                       READIT
  144 IF(72-I)49,145,146
  145 ZEXP=NUMB1
      GO TO 201
  146 ZEXP=NUMB1
      IF(IS(I+1)-BLANK)163,200,163
  163 IF(IS(I+1))189,64,64
  189 IF(IS(I+1)-ZZ)29,92,29
   62 ZEXP=ZEXP*10.                                                   READIT
      GO TO 201                                                       READIT
   61 IF(IS(I+1)-BLANK)63,200,63                                      READIT
   63 IF(IS(I+1))189,62,64
   64 TEMP=IS(I+1)/1073741824
      ZEXP=10.*ZEXP+TEMP                                              READIT
      GO TO 200                                                       READIT
 CHECK FOR Z                                                          READIT
   89 IF(IS(I)-ZZ)29,92,29
   92 IF(NUM)93,90,93
   93 CALL NUMBER(KB)
   90 KB=1                                                            READIT
      GO TO 34                                                        READIT
 CHECK FOR PLUS                                                       READIT
   42 IF(IS(I)-PLUS)72,73,72                                          READIT
C  PLUS OR E FOUND - SET EXPONENT                                     READIT
   73 IGN=1                                                           READIT
      GO TO 21                                                        READIT
 CHECK FOR E, IT WAS NOT PLUS                                         READIT
   72 IF(IS(I)-E)74,73,74                                             READIT
 CHECK FOR DECIMAL POINT, IT WAS NOT E                                READIT
   74 IF(IS(I)-DOT)76,75,76                                           READIT
C  DECIMAL - SET VALUE                                                READIT
   75 DEC=1.                                                          READIT
      GO TO 21                                                        READIT
 CHECK FOR NUMBER, NOT DECIMAL                                        READIT
   76 IF(IS(I)-10737418240)77,29,89
 CHECK FOR TYPE OF NUMBER AND SET ACCORDINGLY                         READIT
   77 IF(IGN)144,177,144
  177 IF(DEC)78,79,80
   78 MACH=77                                                         READIT
      GO TO 100                                                       READIT
   79 NUMB=10*NUMB+NUMB1
      GO TO 21                                                        READIT
   80 DEC=DEC*10.                                                     READIT
      TEMP=NUMB1
      DECP=DECP+TEMP/DEC                                              READIT
   21 CONTINUE                                                        READIT
      IF(NUM)201,303,201
  201 CALL NUMBER(KB)                                                 READIT
  303 IF(LERR)18,18,32
   32 WRITE(IOUT,420)                                                 READIT
  420 FORMAT(1H1)                                                     READIT
```

```
      DO 21 I=IA,72                                                     READIT
      ISS(I)=IS(I)                                                      READIT
      NUMB1=IS(I)/1073741824
CHECK FOR PROCESSING A NUMBER                                           READIT
   22 IF(NUM)25,24,25                                                   READIT
CHECK FOR BLANK - NOT A NUMBER                                          READIT
   24 IF(IS(I)-BLANK)26,21,26                                           READIT
CHECK FOR MINUS - NOT BLANK, ITE=-1 IF -0, =0 IF +0, = +1 OTHERWISE     READIT
   26 IG=IS(I)
      CALL CHECK(IG ,ITE)                                               READIT
      IF(ITE)91,21,28
CONTAINS MINUS, SET SIGN                                                READIT
   91 IF(IS(I)-ZZ)29,92,30
   30 NUM=-1                                                            READIT
      GO TO 21                                                          READIT
CHECK FOR + OR -                                                        READIT
   28 IF(IS(I))89,728,628                                               READIT
  728 MACH=28                                                           READIT
      GO TO 100                                                         READIT
CHECK FOR LEGALITY OF POSITIVE IS, FIRST DECIMAL POINT                  READIT
  628 IF(IS(I)-DOT)36,37,29                                             READIT
   29 IG=I                                                              READIT
      DO 327 IH=IG,72                                                   READIT
  327 ISS(IH)=IS(IH)                                                    READIT
      WRITE(IOUT,500)(ISS(IH),IH=1,72)
      WRITE(IOUT,31)                                                    READIT
   31 FORMAT(50H0YOU GOOFED - ILLEGAL CHARACTER ON PRECEEDING CARD)     READIT
CHECK FOR NEW PROBLEM                                                   READIT
      GO TO 32                                                          READIT
   37 DEC=1.                                                            READIT
      NUM=1                                                             READIT
      GO TO 21                                                          READIT
CHECK NEXT NUMBER                                                       READIT
   36 IF(NUMB1-10)33,29,38
   38 IF(IS(I)-PLUS)29,21,29                                            READIT
CONTROL SET AND STORAGE OF LEGITIMATE NUMBER                            READIT
   33 NUM=1                                                             READIT
      NUMB=NUMB1
      GO TO 21                                                          READIT
CHECK FOR BLANK IN NUMBER BEING PROCESSED                               READIT
   25 IF(IS(I)-BLANK)89,34,35                                           READIT
C  BLANK FOUND - END OF NUMBER                                          READIT
  200 CONTINUE                                                          READIT
   34 CALL NUMBER(KB)                                                   READIT
  301 NUM=0                                                             READIT
      IGN=0                                                             READIT
      DEC=0.                                                            READIT
      NUMB=0                                                            READIT
      KB=0                                                              READIT
      ZEXP=0.                                                           READIT
      DECP=0.                                                           READIT
      GO TO 21                                                          READIT
CHECK FOR MINUS, IT WAS NOT BLANK                                       READIT
   35 CALL CHECK(IS(I),ITE)                                             READIT
      IF(ITE)40,41,42                                                   READIT
C  MINUS FOUND - SET AS EXPONENT                                        READIT
   40 IGN=-1                                                            READIT
      GO TO 21                                                          READIT
CHECK FOR EXPONENT SINCE IS CONTAINS ZERO                               READIT
   41 IF(IGN)44,45,44                                                   READIT
CHECK FOR DECIMAL - IT WAS NOT EXPONENT                                 READIT
```

```
   23 I3(J3)=I                                                          NUMBER
      IF(I)19,24,25                                                     NUMBER
   24 N3(J3)=NUMB                                                       NUMBER
      GO TO 19                                                          NUMBER
   25 P3(J3)=PNUMB                                                      NUMBER
      GO TO 19                                                          NUMBER
   14 J4=J4+1                                                           NUMBER
   26 IF(731-J4)102,102,27                                              NUMBER
   27 I4(J4)=I                                                          NUMBER
      IF(I)19,28,29                                                     NUMBER
   28 N4(J4)=NUMB                                                       NUMBER
      GO TO 19                                                          NUMBER
   29 P4(J4)=PNUMB                                                      NUMBER
      GO TO 19                                                          NUMBER
   15 J5=J5+1                                                           NUMBER
   30 IF(731-J5)102,102,31                                              NUMBER
   31 I5(J5)=I                                                          NUMBER
      IF(I)19,32,33                                                     NUMBER
   32 N5(J5)=NUMB                                                       NUMBER
      GO TO 19                                                          NUMBER
   33 P5(J5)=PNUMB                                                      NUMBER
      GO TO 19                                                          NUMBER
   16 J6=J6+1                                                           NUMBER
   34 IF(601-J6)102,102,35                                              NUMBER
   35 I6(J6)=I                                                          NUMBER
      IF(I)19,36,37                                                     NUMBER
   36 N6(J6)=NUMB                                                       NUMBER
      GO TO 19                                                          NUMBER
   37 P6(J6)=PNUMB                                                      NUMBER
      GO TO 19                                                          NUMBER
      END                                                               NUMBER
```

```
$IBFTC FPMAP    LIST,DECK,REF,DD,XR7
      SUBROUTINE FPMAP(CENX,CENY,NOD,XPRNT,YPRNT,IX,L2)
      COMMON Z
      DIMENSION Z(20000),IB(100),IE(100),XITEM(2),CENX(IX,L2),
     1CENY(IX,L2),ZA(200),ZTE(100   ),XPRNT(IX,L2),YPRNT(IX,L2),
     2PRNT(113    ),NOD(IX,L2)
      EQUIVALENCE (Z(6116),IB(1)),(Z(6216),IE(1)),(Z(5200),LB),
     1(ZA(166),XITEM(1)),
     2(Z(627),ZA(1)),(Z(2),IOUT),(Z(11),LERR),
     3(Z(17316),ZTE(1)),
     4(Z(5425),UC),(Z(5426),VC),
     5(Z(5431    ),ITEMP),(Z(5427     ),RS),(Z(5418),THE),(Z(5417),IJ),
     6(ZA(  2   ),Y),
     7                  (Z(5415),II),(Z(5429 ),XS),(Z(5430 ),YS),
     8(Z(5414),DY),(ZA(1    ),X),(Z(5413),DX)
      EQUIVALENCE                   (Z(5716    ),PRNT(1)),(Z(5410),B),
     1(Z(5409),D),(Z(5408),C),(Z(5407),G),(Z(5406),LINE),
     2(Z(5405),LU),(Z(5404),LL),(Z(5403),I),(Z(5402),IC),(Z(5401),IA),
     3(Z(5400),J),(Z(5428),NT),(Z(5432),NODA),(Z(5211),DELX),
     4(ZA(4),UO),(ZA(5),VO),(ZA(6),U),(ZA(7),V)
      DATA  LANK,JPR,MINU,MINI/1H ,1HX,1H-,1HI/
      INTEGER      PRNT,XPRNT,YPRNT,XPRN,YPRN
   12 FORMAT(1H1)
   10 FORMAT(1H ,112A1,I4)
   11 FORMAT( 32H1COLLECTOR AND RADIATOR NODE MAP/112H0 123456789111111
     1111122222222223333333333444444444455555555555666666666677777777778
     28888888889999999999111111111/11X,9(10H0123456789),11H00000000001/1
     301X,11H01234567890)
   13 FORMAT(1H0,5X,3HMAP,12X,3HX-Y,18X,                 14X,4HNODE)
  235 FORMAT(12H COORDINATES,5X,11HCOORDINATES,12X,                  6H
     1NUMBER)
   30 FORMAT(3X,4HNODE,13X,5HX-Y-Z)
  234 FORMAT(8H  NUMBER,10X,11HCOORDINATES)
   32 FORMAT(I6,3E12.5)
   18 FORMAT(2I6,2E12.4,I7)
      RSO=0.
      RS=0.
      SSO=0.
      SS=0.
      XITEM(2)=0.
      IMAX=XITEM(1)
      ZTE(1)=DELX/2.
      DO 801  I=2,IMAX
      ZTE(I)=ZTE(I-1)+DELX
  801 CONTINUE
      NODA=XITEM(1)+XITEM(2)
      NT  = 1+NODA
      DO 1020 J=1,LB
      IA=IB(J)
      IC=IE(J)
      DO 1021  I=IA,IC
      NOD(I,J)= NT
 1021 NT=NT+1
 1020 CONTINUE
      NT=NT+1
      LL=LB+1
      LU=2*LB
      DO 1022  J=LL,LU
      IA= IB(J)
      IC= IE(J)
      DO 1023 I=IA,IC
```

```
      NOD(I,J)=NT
 1023 NT=NT+1
 1022 CONTINUE
  111 WRITE(IOUT,11)
      LINE=5
      G=ABS((VO-V)/10.)
      C=(VO+V)/2.
      D=(UO+U)/2.
      B=ATAN(C/D)*57.296
      DO 24 I=1,111
      PRNT(I+1)=MINU
   24 CONTINUE
   26 PRNT(1)=MINU
      PRNT(113)=0
      WRITE (IOUT,10)(PRNT(II),II=1,113)
      PRNT(113)=1
   27 PRNT(1)=MINI
      PRNT(112)=MINI
      DX=X/110.
      DY=1.71*DX
      YS=Y+DY
      IJ=1
  112 YS=YS-DY
      XS=-DX
      II=1
  103 XS=XS+DX
      J=1
  102 I=IB(J)
  101 IF(ABS(YS-CENY(I,J))-DY)1201,201,200
 1201 IF(YS.GE.CENY(I,J)) GO TO 201
  200 PRNT(II+1)= LANK
      GO TO 1
  201 IF(ABS(XS-CENX(I,J))-DX)1202,202,200
 1202 IF(XS.LE.CENX(I,J)) GO TO 202
      GO TO 200
  202 PRNT(II+1)=JPR
      XPRNT(I,J)=II
      YPRNT(I,J)=IJ
      GO TO 3
    1 IF(I-IE(J))104,2,2
  104 I=I+1
      GO TO 101
    2 IF(J-LB)105,3,3
  105 J=J+1
      GO TO 102
    3 IF(II-110)106,107,107
  106 II=II+1
      GO TO 103
  107 WRITE(IOUT,10)(PRNT(II),II=1,113)
      PRNT(113)=PRNT(113)+1
      IF(YS)4,4,108
  108 IF(LINE-55)110,109,109
  110 LINE=LINE+1
  113 IJ=IJ+1
      GO TO 112
  109 WRITE(IOUT,11)
      LINE=5
      GO TO 113
    4 I=1
      DO 339 I=1,112
  339 PRNT(I)=MINU
```

```
      PRNT(113)=0
      WRITE(IOUT,10)(PRNT(I),I=1,113)
   41 IJJ=IJ
      IF(LINE-30)124,123,123
  123 WRITE(IOUT,12)
      LINE=0
  124 WRITE(IOUT,13)
      WRITE(IOUT,235)
      LINE=LINE+3
      IJ=1
  122 II=1
  121 J=1
  120 I=IB(J)
  119 IF(XPRNT(I,J)-II)16,116,16
  116 IF(YPRNT(I,J)-IJ)16,117,16
  117 IF(ABS(CENX(I,J)-D)-1.0E-4)118,118,156
  156 THE=ATAN((CENY(I,J)-C)/(CENX(I,J)-D))*57.296-B
      IF(CENX(I,J)-D)321,321,322
  321 IF(CENY(I,J)-C)328,330,324
  330 IF(CENX(I,J)-D)324,324,131
  324 THE=THE+360.
      GO TO 323
  328 IF(THE)324,323,323
  322 IF(CENY(I,J)-C)340,330,340
  340 THE=THE+180.
  323 GO TO 315
  118 THE=ATAN(D/C)*57.296
  130 IF(CENY(I,J)-C)315,131,131
  131 THE=180.+THE
  315 CONTINUE
   15 THEP=THE/57.296
      IF(ABS(CENY(I,J)-C)-G)150,150,151
  150 RS=(CENX(I,J)-D)/COS(THEP+B)
      GO TO 17
  151 RS=(CENY(I,J)-C)/SIN(THEP+B)
      RS=ABS(RS)
   17 CONTINUE
      IF(LINE-55)153,152,152
  152 WRITE(IOUT,12)
      WRITE(IOUT,13)
      WRITE(IOUT,235)
      LINE=2
  153 WRITE(IOUT,18)II,IJ,CENX(I,J),CENY(I,J),        NOD(I,J)
      LINE=LINE+1
      GO TO 158
   16 IF(IE(J)-I)155,155,154
  154 I=I+1
      GO TO 119
  155 IF(LB-J)158,158,157
  157 J=J+1
      GO TO 120
  158 IF(110-II)159,159,160
  160 II=II+1
      GO TO 121
  159 IF(IJJ-IJ)23,23,161
  161 IJ=IJ+1
      GO TO 122
   23 IF(LINE-30)162,163,163
  163 LINE=0
      WRITE(IOUT,12)
  162 WRITE(IOUT,31)
```

```
      WRITE(IOUT,30)
      WRITE(IOUT,234)
      LINE=LINE+4
      I=1
      ITEMP=XITEM(1)
      UC=(UO+U)/2.
      VC=(VO+V)/2.
      RC=(RSO+RS)/2.
      SC=(SSO+SS)/2.
  133 IF(LINE-55)34,165,165
  165 WRITE(IOUT,12)
      WRITE(IOUT,30)
      WRITE(IOUT,234)
      LINE=2
   34 WRITE(IOUT,32)I,UC,VC,ZTE(I)
      IF(ITEMP-I)167,167,166
  166 I=I+1
      GO TO 133
  167 IF(XITEM(2))39,39,36
   36 ITEMP=XITEM(1)+XITEM(2)
      I=XITEM(1)+1.
  135 IF(LINE-55)37,168,168
  168 WRITE(IOUT,12)
      WRITE(IOUT,30)
      WRITE(IOUT,234)
      LINE=2
   37 WRITE(IOUT,32)I,RC,SC,ZTE(I)
      IF(ITEMP-I)39,39,169
  169 I=I+1
      GO TO 135
   31 FORMAT( 31H0 THEMOELEMENT MESH INFORMATION)
   39 RETURN
      END
```

```
$IBFTC PRO2     LIST,DECK,DD,XR7
      SUBROUTINE PRO2                                                   PRO2
C                                                                       PRO2
      COMMON Z                                                          PRO2
      EQUIVALENCE (Z(2),IOUT),(Z(11),LERR),(Z(3575),I2(1)),(Z(3375),N2( PRO2
     21)),              (Z(614),M1),(Z(615),M2),(Z(616),M3),(Z(617),M4), PRO2
     3(Z(618),M5),(Z(619),M6),(Z(620),M7),(Z(621),M8),(Z(622),M9),(Z(623 PRO2
     4),M10),(Z(624),M11),(Z(625),M12),(Z(626),CTH),(Z(627),ZA(1)),(Z(82 PRO2
     58),J2),(Z(3375),P2(1)),(Z(614),MA2(1)),(Z(3763),MS(1)),(Z(3762),I) PRO2
     6,(Z(3761),N)                  ,(Z(3760),NA),(Z(3759),NZ)
      DIMENSION Z(16000)                                                PRO2
      DIMENSION I2(200),N2(200),P2(200),ZA(200),MS(12),MA2(12)          PRO2
      NAMELIST /NAM1/CTH/NAM2/ZA
      M6=0
      DO 170 I=1,12                                                     PRO2
      MS(I)=MA2(I)                                                      PRO2
  170 MA2(I)=0                                                          PRO2
      LERR=0                                                            PRO2
      I=1                                                               PRO2
   17 IF(I2(I))13,14,31                                                 PRO2
   13 WRITE(IOUT,15)                                                    PRO2
   15 FORMAT(27H0ILLEGAL CONTROL IN *2 DATA)                            PRO2
   30 LERR=1
   25 RETURN                                                            PRO2
   14 IF(N2(I)-1)13,1,16                                                PRO2
   16 IF(12-N2(I))13,12,18
   18 N=N2(I)                                                           PRO2
   34 GO TO (100,2,3,4,5,6,7,8,9,10,11,12),N                            PRO2
  100 WRITE(IOUT,101)                                                   PRO2
  101 FORMAT (32H0MACHINE ERROR - SUBROUTINE PRO2)                      PRO2
      GO TO 30
    1 I=I+1                                                             PRO2
   36 IF(I2(I))19,20,21                                                 PRO2
   19 IF(I-J2)22,26,27                                                  PRO2
   22 I=I+1                                                             PRO2
   24 IF(I2(I))19,14,31                                                 PRO2
   20 CTH=N2(I)                                                         PRO2
      M1=1                                                              PRO2
      GO TO 19                                                          PRO2
   21 CTH=P2(I)                                                         PRO2
      M1=1                                                              PRO2
      GO TO 19                                                          PRO2
  175 IF(M1)100,27,28                                                   PRO2
   27 WRITE(IOUT,29)                                                    PRO2
   29 FORMAT(43H0INSUFFICIENT *2 DATA - I CANNOT DO P-OBLEM)            PRO2
      GO TO 30                                                          PRO2
   31 IF(P2(I)-1.)13,1,32                                               PRO2
   32 IF(12.-P2(I))13,12,33                                             PRO2
   33 N=P2(I)                                                           PRO2
      TEMP=N                                                            PRO2
   35 IF(P2(I)-TEMP)13,34,13                                            PRO2
   28 IF(N2)100,27,37                                                   PRO2
   37 IF(M3)100,39,38                                                   PRO2
   39 IF(M4)100,27,40                                                   PRO2
   38 IF(M4)100,40,41                                                   PRO2
   41 WRITE(IOUT,42)                                                    PRO2
   42 FORMAT(41H0TOO MUCH *2 DATA - IT CONTRADICTS - GOOF)              PRO2
      GO TO 30                                                          PRO2
```

```
   40 IF(M5)100,43,44                                                     PR02
   43 IF(M6)100,27,45                                                     PR02
   44 IF(M6)100,45,41                                                     PR02
   45 IF(M7)100,27,46                                                     PR02
   46 IF(M8)100,27,47                                                     PR02
   47 IF(M9)100,27,48                                                     PR02
   48 IF(M10)100,27,49                                                    PR02
   49 IF(M11)100,27,79                                                    PR02
   79 IF(M12)100,27,200
  200 IF(M6)80,201,80
  201 M6=M60
   80 IF(M7 -M6 )67,81,67
   81 IF(M9-M6 )67,82,67
   82 IF(M10-M6 )67,83,67
   83 IF(M3)100,116,115
  115 IF(8+M6-M11)67,117,67                                               PR02
  116 IF(12+M6-M11)67,117,67                                              PR02
  117 IF(M11-1000)118,119,119                                             PR02
  119 WRITE(IOUT,120)                                                     PR02
  120 FORMAT(31H0TOO MANY MESH POINTS SPECIFIED)                          PR02
      GO TO 30                                                            PR02
  118 CONTINUE
      RETURN
    2 NA=1                                                                PR02
      NZ=3                                                                PR02
      M2=1                                                                PR02
   59 DO 53 II=NA,NZ                                                      PR02
      I=I+1                                                               PR02
   57 IF(I2(I))54,55,56                                                   PR02
   55 ZA(II)=N2(I)                                                        PR02
      GO TO 53                                                            PR02
   56 ZA(II)=P2(I)                                                        PR02
      GO TO 53                                                            PR02
   54 WRITE(IOUT,58)                                                      PR02
   58 FORMAT(20H0IMPROPER Z LOCATION)                                     PR02
      GO TO 30                                                            PR02
   53 CONTINUE                                                            PR02
      GO TO 19                                                            PR02
    3 NA=4                                                                PR02
      NZ=7                                                                PR02
      M3=1                                                                PR02
      M60=1
      GO TO 59                                                            PR02
    4 NA=4                                                                PR02
      NZ=11                                                               PR02
      M4=1                                                                PR02
      M60=2
      GO TO 59                                                            PR02
    5 NA=12                                                               PR02
      IF(I2(I+2))60,61,61
   60 NZ=12                                                               PR02
      M5=1                                                                PR02
      GO TO 59                                                            PR02
   61 NZ=13                                                               PR02
      M5=2                                                                PR02
      GO TO 59                                                            PR02
    6 NA=14                                                               PR02
      DO 62 II=1,40                                                       PR02
      JA=I+II+1                                                           PR02
      IF(I2(JA))63,62,62                                                  PR02
   63 NZ=NA+II-1                                                          PR02
```

```
      M6=(NZ-13)/2
      GO TO 59
   62 CONTINUE
      M6=20
      GO TO 59
    7 NA=54
      DO 64 II=1,20
      JA=I+II+1
      IF(I2(JA))65,64,64
   65 NZ=NA+II-1
      M7=NZ-53
      GO TO 59
   64 CONTINUE
      M7=20
      GO TO 59
   67 WRITE(IOUT,68)
   68 FORMAT(31H0INCONSISTENT *2 DATA - TCH TCH)
      GO TO 30
    8 M8=1
      NA=75
      NZ=77
      GO TO 59
    9 NA=78
      DO 69 II=1,40
      JA=I+II+1
      IF(I2(JA))70,69,69
   70 NZ=NA+II-1
      M9=(NZ-77)/2
      GO TO 59
   69 CONTINUE
      M9=20
      GO TO 59
   10 NA=118
      DO 72 II=1,40
      JA=I+II+1
      IF(I2(JA))73,72,72
   73 NZ=NA+II-1
      M10=(NZ-117)/2
      GO TO 59
   72 CONTINUE
      M10=20
      GO TO 59
   11 NA=158
      DO 90 II=1,32
      JA=I+II+1
      IF(I2(JA))91,90,90
   91 NZ=NA+II-1
      M11=NZ-157
      GO TO 59
   90 CONTINUE
      M11=32
      GO TO 59
   12 NA=190
      NZ=193
      M12=1
      GO TO 59
   26 IF(M1)100,171,172
  171 M1=MS(1)
  172 IF(M2)100,173,174
  173 M2=MS(2)
  174 IF(M3)100,176,179
```

```
176 IF(M4)100,178,179                                                   PRO2
178 M3=MS(3)                                                            PRO2
.   M4=MS(4)                                                            PRO2
179 DO 180 I=5,12                                                       PRO2
182 IF(MA2(I))100,181,180                                               PRO2
181 MA2(I)=MS(I)                                                        PRO2
180 CONTINUE                                                            PRO2
    GO TO 175                                                           PRO2
    END                                                                 PRO2
$ORIGIN          ALPHA
```

```
$IBFTC PRO3     LIST,DECK,DD,XR7
      SUBROUTINE PRO3                                                   PRO3
      COMMON Z
      DIMENSION Z(20000),TES(200),ALPHAS(200),THERMS(200),RESIS(200),   PRO3
     2CAPS(200),DENS(200),EPSIS(200),ICS(200),IZC(10),ITAB(10),IZD(10), PRO3
     3I3(1600),P3(1600),N3(1600),IC(200),TE(200),ALPHA(200),THERM(200), PRO3
     4RESIST(200),CAP(200),DEN(200),EPSI(200),IZA(200)                  PRO3
      DIMENSION ZA(200),IXB(20),IXE(20),MATTE(15)
      EQUIVALENCE (Z(12),IPR),(Z(832),MACH),(Z(11),LERR),(Z(833),TES(1))PRO3
     2,(Z(1033),ALPHAS(1)),(Z(1233),THERMS(1)),(Z(1433),RESIS(1)),(Z(   PRO3
     31633),CAPS(1)),(Z(1833),DENS(1)),(Z(2033),EPSIS(1)),(Z(2233),ICS( PRO3
     41)),(Z(2433),ITAB(1)),(Z(2443),NP3),(Z(8589),IZC(1)),(Z(8579),    PRO3
     5IZD(1)),(Z(8379),IC(1)),(Z(8179),TE(1)),(Z(7979),ALPHA(1)),(Z(7779PRO3
     6),THERM(1)),(Z(7579),RESIST(1)),(Z(7379),CAP(1)),(Z(7179),DEN(1)),PRO3
     7(Z(6979),EPSI(1)),(Z(5379),I3(1)),(Z(3779),P3(1)),(Z(3779),N3(1))
     8,(Z(3778),I),(Z(3777),N),(Z(3776),TEMP),(Z(3775),IP),(Z(829 ),J3),PRO3
     9(Z(13661),IZA(1)),(Z(2),IOUT)
      EQUIVALENCE (Z(17416),IXB(1)),(Z(17436),IXE(1)),(Z(627),ZA(1)),
     1(ZA(54),MATTE(1))
      NAMELIST/NAM1/TES/NAM2/ALPHAS/NAM3/THERMS/NAM4/RESIS/NAM5/CAPS/
     1NAM6/DENS/NAM7/EPSIS/NAM8/ICS
      LERR=0                                                            PRO3
      IPRO=IPR                                                          PRO3
      IPR=1                                                             PRO3
    1 IF(NP3)2,205,4
    2 MACH=1                                                            PRO3
  100 WRITE(IOUT,101) MACH
  101 FORMAT(32H0MACHINE ERROR, PRO3, STATEMENT I4)                     PRO3
      GO TO 8
    4 DO 5 I=1,IPRO                                                     PRO3
      Z(I+9799)=TES(I)                                                  PRO3
      Z(I+9599)=ALPHAS(I)                                               PRO3
      Z(I+9399)=THERMS(I)                                               PRO3
      Z(I+9199)=RESIS(I)                                                PRO3
      Z(I+8999)=CAPS(I)                                                 PRO3
      Z(I+8799)=DENS(I)                                                 PRO3
      Z(I+8599)=EPSIS(I)                                                PRO3
    5 IZA(I)=ICS(I)                                                     PRO3
  205 DO 94 I=1,6
      IZC(I)=ITAB(I)                                                    PRO3
      IZD(I)=0                                                          PRO3
   94 ITAB(I)=0                                                         PRO3
    3 I=1                                                               PRO3
   21 IF(I3(I))6,15,10                                                  PRO3
    6 WRITE(IOUT,7)                                                     PRO3
    7 FORMAT(20H0ILLEGAL CONTROL, *3)
    8 LERR=1
    9 RETURN                                                            PRO3
   10 IF(P3(I)-1.)6,11,12                                               PRO3
   12 IF(P3(I)- 6.)13,11,6
   13 N=P3(I) +.01                                                      PRO3
      TEMP=N                                                            PRO3
      IF(ABS(P3(I)-TEMP)-.01)110,110,6
   11 N=P3(I)                                                           PRO3
  110 IC(IPR)=N                                                         PRO3
      GO TO 14                                                          PRO3
   15 IF(N3(I)-1)6,16,17                                                PRO3
   17 IF(N3(I)- 6)16,16,6
```

```
   16 IC(IPR)=N3(I)                                                    PRO3
   14 IP=1                                                             PRO3
      I=I+1                                                            PRO3
   61 IF(I-J3)19,70,60                                                 PRO3
   19 IF(I3(I))20,22,31                                                PRO3
   20 I=I+1                                                            PRO3
      IC(IPR-1)=-1
      GO TO 21                                                         PRO3
   60 MACH=61                                                          PRO3
      GO TO 100                                                        PRO3
   22 P3(I)=N3(I)                                                      PRO3
   24 GO TO (31,32,33,34,35,36,37),IP                                  PRO3
   31 TE(IPR)=P3(I)                                                    PRO3
   26 IP=IP+1                                                          PRO3
      I=I+1                                                            PRO3
      IF(I3(I))23,22,24                                                PRO3
   23 WRITE(IOUT,25)                                                   PRO3
   25 FORMAT(12HOBAD *3 DATA)                                          PRO3
      GO TO 8                                                          PRO3
   32 ALPHA(IPR)=P3(I)                                                 PRO3
      GO TO 26                                                         PRO3
   33 THERM(IPR)=P3(I)                                                 PRO3
      GO TO 26                                                         PRO3
   34 RESIST(IPR)=P3(I)                                                PRO3
      GO TO 26                                                         PRO3
   35 CAP(IPR)=P3(I)                                                   PRO3
      GO TO 26                                                         PRO3
   36 DEN(IPR)=P3(I)                                                   PRO3
      CAP(IPR)=CAP(IPR)*DEN(IPR)
      GO TO 26                                                         PRO3
   37 EPSI(IPR)=P3(I)                                                  PRO3
      IPR=IPR+1                                                        PRO3
      IC(IPR)=0                                                        PRO3
      GO TO 14                                                         PRO3
   70 IC(IPR-1)=-1
      IPR=IPR-1
      IH=2                                                             PRO3
      IM=1                                                             PRO3
      IG=1                                                             PRO3
      TMIN=TE(1)                                                       PRO3
   83 DO 42 I=IH,IPR                                                   PRO3
   43 IF(IC(I))42,42,45
   45 IJ=I-1                                                           PRO3
      GO TO 51
   42 CONTINUE                                                         PRO3
      IJ=IPR                                                           PRO3
   51 DO 40 I=IH,IJ                                                    PRO3
      IF(TMIN-TE(I))40,23,41
   41 TMIN=TE(I)                                                       PRO3
      IM=I                                                             PRO3
   40 CONTINUE                                                         PRO3
   46 IF(IM-IG)47,50,52                                                PRO3
   47 MACH=46                                                          PRO3
      GO TO 100                                                        PRO3
   50 IG=IG+1                                                          PRO3
  183 IF(IG-IJ)80,81,182
   80 IH=IG+1                                                          PRO3
      TMIN=TE(IG)                                                      PRO3
      IM=IG                                                            PRO3
      GO TO 51                                                         PRO3
  182 MACH=183
```

```
      GO TO 100                                                          PRO3
   52 TEMP=TE(IG)                                                        PRO3
      TE(IG)=TE(IM)                                                      PRO3
      TE(IM)=TEMP                                                        PRO3
      TEMP=ALPHA(IG)                                                     PRO3
      ALPHA(IG)=ALPHA(IM)                                                PRO3
      ALPHA(IM)=TEMP                                                     PRO3
      TEMP=THERM(IG)                                                     PRO3
      THERM(IG)=THERM(IM)                                                PRO3
      THERM(IM)=TEMP                                                     PRO3
      TEMP=RESIST(IG)                                                    PRO3
      RESIST(IG)=RESIST(IM)                                              PRO3
      RESIST(IM)=TEMP                                                    PRO3
      TEMP=CAP(IG)                                                       PRO3
      CAP(IG)=CAP(IM)                                                    PRO3
      CAP(IM)=CAP(IG)                                                    PRO3
      TEMP=DEN(IG)                                                       PRO3
      DEN(IG)=DEN(IM)                                                    PRO3
      DEN(IM)=TEMP                                                       PRO3
      TEMP=EPSI(IG)                                                      PRO3
      EPSI(IG)=EPSI(IM)                                                  PRO3
      EPSI(IM)=TEMP                                                      PRO3
      GO TO 50                                                           PRO3
   81 IF(IJ-IPR)82,85,84                                                 PRO3
   82 IG=IG+1
      IH=IG+1
      IM=IG
      TMIN=TE(IM)                                                        PRO3
      GO TO 83                                                           PRO3
   84 MACH=81                                                            PRO3
      GO TO 100                                                          PRO3
   85 CONTINUE
C  ALL INFO IN TABLES AND ARRANGED IN INCREASING TEMPERATURE ORDER       PRO3
C  NOW PICK OUT TABLE NUMBERS                                            PRO3
      DO 87 I=1,IPR                                                      PRO3
      IF(IC(I))87,87,88                                                  PRO3
   88 IM=IC(I)                                                           PRO3
   89 IF(IZD(IM))90,91,92                                                PRO3
   90 MACH=89                                                            PRO3
      GO TO 100                                                          PRO3
   91 IZD(IM)=I                                                          PRO3
      GO TO 87                                                           PRO3
   92 WRITE(IOUT,93)                                                     PRO3
   93 FORMAT(27H0DUPLICATE TABLE NUMBER, *3)
      GO TO 23                                                           PRO3
   87 CONTINUE                                                           PRO3
C  MERGE TABLES TO COMPLETED FORM                                        PRO3
      IA=1                                                               PRO3
      DO 95 I=1,6
      IH=1                                                               PRO3
      IF(IZC(I))96,97,98                                                 PRO3
   96 MACH=95                                                            PRO3
      GO TO 100                                                          PRO3
   97 IF(IZD(I))96,95,99                                                 PRO3
   99 IG=IZD(I)                                                          PRO3
      ITAB(I)=IA                                                         PRO3
  104 TES(IA)=TE(IG)                                                     PRO3
      ALPHAS(IA)=ALPHA(IG)                                               PRO3
      THERMS(IA)=THERM(IG)                                               PRO3
      RESIS(IA)=RESIST(IG)                                               PRO3
      CAPS(IA)=CAP(IG)                                                   PRO3
```

```
      DENS(IA)=DEN(IG)                                                  PRO3
      EPSIS(IA)=EPSI(IG)                                                PRO3
      ICS(IA)=IC(IG)                                                    PRO3
      GO TO (102,103,107),IH                                            PRO3
  102 IH=2                                                              PRO3
      IA=IA+1                                                           PRO3
      IG=IG+1                                                           PRO3
      GO TO 104                                                         PRO3
  103 IA=IA+1                                                           PRO3
      IG=IG+1                                                           PRO3
      IF(IC(IG))105,104,107                                             PRO3
  105 IH=3                                                              PRO3
      GO TO 104                                                         PRO3
   98 IF(IZD(I))96,108,99                                               PRO3
  108 IG=IZC(I)                                                         PRO3
      ITAB(I)=IA                                                        PRO3
  114 TES(IA)=Z(IG+9799)                                                PRO3
      ALPHAS(IA)=Z(IG+9599)                                             PRO3
      THERMS(IA)=Z(IG+9399)                                             PRO3
      RESIS(IA)=Z(IG+9199)                                              PRO3
      CAPS(IA)=Z(IG+8999)                                               PRO3
      DENS(IA)=Z(IG+8799)                                               PRO3
      EPSIS(IA)=Z(IG+8599)                                              PRO3
      ICS(IA)=IZA(IG)                                                   PRO3
      GO TO (112,113,107),IH                                            PRO3
  112 IH=2                                                              PRO3
      IA=IA+1                                                           PRO3
      IG=IG+1                                                           PRO3
      GO TO 114                                                         PRO3
  113 IA=IA+1                                                           PRO3
      IG=IG+1                                                           PRO3
      IF(IC(IG))115,114,107                                             PRO3
  115 IH=3                                                              PRO3
      GO TO 114                                                         PRO3
  107 IA=IA+1                                                           PRO3
   95 CONTINUE                                                          PRO3
      IA=IA-1                                                           PRO3
      IF(200-IA)117,118,118                                             PRO3
  117 WRITE(IOUT,119) IA
  119 FORMAT(I4,17HTABLE ENTRIES, *3)
      GO TO 23                                                          PRO3
  118 IPR=IA                                                            PRO3
      NP3=1                                                             PRO3
      IM=1
      NSG=1
 1001 IX=1
 1006 XXX=ICS(IX)
      NN=53+IM
      IF(ABS(XXX-ZA(NN    ))-1.E-5)1003,1003,1002
 1002 IF(IX-200)1004,1005,1005
 1004 IX=IX+1
      GO TO 1006
 1005 WRITE(IOUT,1007)
 1007 FORMAT(21H0 TE MATERIAL MISSING)
      GO TO 8
 1003 NONIE=ICS(IX)
      IXB(NONIE)=IX
 1008 IF(ICS(IX))1010,2081,2081
 2081 IF(IX-200)1009,1010,1010
 1009 IX=IX+1
      GO TO 1008
```

```
1010 IXE(NONIE)=IX
     NSG=NSG-1
     IM=IM+1
     IF(NSG)1001,1011,1001
1011 CONTINUE
     RETURN                                                         PRO3
     END
```

```
$IBFTC PRO4    LIST,DECK,REF,DD,XR7
      SUBROUTINE PRO4                                                   PRO4
      COMMON Z                                                          PRO4
      DIMENSION Z(20000),CONTRA(200),EMISIA(200),TEA(200),ALPHAA(200),  PRO4
     2THERMA(200),RESIA(200),CAPA(200),DENA(200),EMISOA(200),IZA(200),  PRO4
     3ICA(200),IZC(30),ITABA(30),IZD(30),I4(1830),P4(1830),N4(1830),    PRO4
     4TE(200),THERM(200),CAP(200),DEN(200),RESIST(200),ALPHA(200),EMISO PRO4
     5(200),EMISI(200),CONTR(200),IC(200)
      DIMENSION IXB(20),IXE(20),ZA(200)
      EQUIVALENCE (Z(832),MACH),(Z(11),LERR),(Z(14),CONTRA(1)),(Z(214), PRO4
     2EMISIA(1)),(Z(414),TEA(1)),(Z(2446),ALPHAA(1)),(Z(2646),THERMA(1)) PRO4
     3,(Z(2846),RESIA(1)),(Z(3046),CAPA(1)),(DENA(1    ),Z(3246)),(Z(3446 PRO4
     4),EMISOA(1)),(Z(5000 ),ICA(1)),(Z(8169),IZA(1)),(Z(8139),IZC(1)), PRO4
     5(Z(8109),IZD(1)),(Z(15940),ITABA(1)),(Z(10000),I4(1)),(Z(11830),  PRO4
     6P4(1)),(Z(11830),N4(1)),(Z(8108),I),(Z(8107),N),(Z(8106),TEMP),   PRO4
     7(Z(8105),IPRO),(Z(662),IPR4),(Z(663),NP4),(Z(8104),IP),(Z(8103),  PRO4
     8NTEMP),(Z(830   ),J4),(Z(7900),THERM(1)),(Z(7700),CAP(1)),(Z(7500), PRO4
     9DEN(1)),(Z(7300),RESIST(1)),(Z(7100),ALPHA(1))                    PRO4
      EQUIVALENCE (Z(6900),EMISO(1)),(Z(6700),EMISI(1)),(Z(6500),CONTR  PRO4
     2(1)),(Z(8102),IH),(Z(8101),IM),(Z(8100),IG),(Z(6499),TMIN),(Z(6299 PRO4
     3),TE(1)),(Z(6298),IJ),(Z(6297),IA),(Z(5997),IC(1)),(Z(2),IOUT)    PRO4
      EQUIVALENCE (Z(4415),MATC),(Z(4416),MATR),(Z(17416),IXB(1)),
     1(Z(17436),IXE(1)),(Z(627),ZA(1)),(ZA(79),MH),(ZA(119),MC)
      NAMELIST/NAM1/ICA/NAM2/TEA/NAM3/THERMA/NAM4/CAPA/NAM5/DENA/
     1NAM6/RESIA/NAM7/CONTRA/NAM8/ALPHAA/NAM9/EMISIA/NAM10/EMISOA
      NAMELIST/NAM11/IXB/NAM12/IXE
      LERR=0                                                            PRO4
      IPRO=IPR4                                                         PRO4
      IPR4=1                                                            PRO4
    1 IF(NP4)2,205,4
    2 MACH=1                                                            PRO4
  100 WRITE(IOUT,101) MACH
  101 FORMAT(32H0MACHINE ERROR, PRO4, STATEMENT I4)                     PRO4
      GO TO 8
    4 DO 5 I=1,IPRO                                                     PRO4
      Z(I+9799)=TEA(I)                                                  PRO4
      Z(I+9599)=ALPHAA(I)                                               PRO4
      Z(I+9399)=THERMA(I)                                               PRO4
      Z(I+9199)=RESIA(I)                                                PRO4
      Z(I+8999)=CAPA(I)                                                 PRO4
      Z(I+8799)=DENA(I)                                                 PRO4
      Z(I+8599)=EMISOA(I)                                               PRO4
      Z(I+8399)=EMISIA(I)                                               PRO4
      Z(I+8199)=CONTRA(I)                                               PRO4
    5 IZA(I)=ICA(I)                                                     PRO4
  205 DO 94 I=1,30
      IZC(I)=ITABA(I)                                                   PRO4
      IZD(I)=0                                                          PRO4
   94 ITABA(I)=0
    3 I=1                                                               PRO4
   21 IF(I4(I))6,15,10                                                  PRO4
    6 WRITE(IOUT,7)                                                     PRO4
    7 FORMAT(20H0ILLEGAL CONTROL, *4)
    8 LERR=1
    9 RETURN                                                            PRO4
   10 IF(P4(I)-1.)6,11,12                                               PRO4
   12 IF(P4(I)-30.)13,11,6                                              PRO4
   13 N=P4(I) +.01
                                                                        PRO4
```

```
110 IC (IPR4)=N
    GO TO 14                                                PRO4
 15 IF(N4(I)-1)6,16,17                                      PRO4
 17 IF(N4(I)-30)16,16,6                                     PRO4
 16 IC (IPR4)=N4(I)
    NTEMP=N4(I)                                             PRO4
 14 IP=1                                                    PRO4
    I=I+1                                                   PRO4
 61 IF(I-J4)19,70,60                                        PRO4
 19 IF(I4(I))20,22,31                                       PRO4
 20 I=I+1                                                   PRO4
    IC(IPR4-1)=-1
    GO TO 21                                                PRO4
 60 MACH=61                                                 PRO4
    GO TO 100                                               PRO4
402 GO TO (31,32,33,34,35,36,37,38),IP                      PRO4
 31 TE(IPR4)=P4(I)                                          PRO4
 26 IP=IP+1                                                 PRO4
    I=I+1                                                   PRO4
    IF(I4(I))23,22,24                                       PRO4
 23 WRITE(IOUT,25)                                          PRO4
 25 FORMAT(12H0BAD *4 DATA)
    GO TO 8                                                 PRO4
 32 THERM(IPR4)=P4(I)                                       PRO4
    GO TO 26                                                PRO4
 33 CAP(IPR4)=P4(I)                                         PRO4
    GO TO 26                                                PRO4
 34 DEN(IPR4)=P4(I)                                         PRO4
    CAP(IPR4)=CAP(IPR4)*DEN(IPR4)
    GO TO 26                                                PRO4
 35 RESIST(IPR4)=P4(I)                                      PRO4
    GO TO 26
403 CONTR(IPR4)=ALPHA(IPR4)                                 PRO4
400 IPR4=IPR4+1
    IC (IPR4)=0
    GO TO 14                                                PRO4
 22 P4(I)=N4(I)                                             PRO4
 24 IF(NTEMP-21)402,403,403                                 PRO4
 36 ALPHA(IPR4)=P4(I)                                       PRO4
    IF(NTEMP-10)403,403,26                                  PRO4
 37 EMISO(IPR4)=P4(I)                                       PRO4
    GO TO 26                                                PRO4
 38 EMISI(IPR4)=P4(I)
    GO TO 400                                               PRO4
 70 IC(IPR4-1)=-1
    IPR4=IPR4-1
    IH=2                                                    PRO4
    IM=1                                                    PRO4
    IG=1                                                    PRO4
    TMIN=TE(1)                                              PRO4
 83 DO 42 I=IH,IPR4                                         PRO4
 43 IF(IC (I))42,42,45
 45 IJ=I-1                                                  PRO4
    GO TO 51
 42 CONTINUE                                                PRO4
    IJ=IPR4                                                 PRO4
 51 DO 40 I=IH,IJ                                           PRO4
    IF(TMIN-TE(I))40,23,41
 41 TMIN=TE(I)                                              PRO4
    IM=I                                                    PRO4
 40 CONTINUE                                                PRO4
```

```
 46 IF(IM-IG)47,50,52                                              PRO4
 47 MACH=46                                                        PRO4
    GO TO 100                                                      PRO4
 50 IG=IG+1                                                        PRO4
183 IF(IG-IJ)80,81,182
 80 IH=IG+1                                                        PRO4
    TMIN=TE(IG)                                                    PRO4
    IM=IG                                                          PRO4
    GO TO 51                                                       PRO4
182 MACH=183
    GO TO 100                                                      PRO4
 52 TEMP=TE(IG)                                                    PRO4
    TE(IG)=TE(IM)                                                  PRO4
    TE(IM)=TEMP                                                    PRO4
    TEMP=THERM(IG)                                                 PRO4
    THERM(IG)=THERM(IM)                                            PRO4
    THERM(IM)=TEMP                                                 PRO4
    TEMP=CAP(IG)                                                   PRO4
    CAP(IG)=CAP(IM)                                                PRO4
    CAP(IM)=TEMP                                                   PRO4
    TEMP=DEN(IG)                                                   PRO4
    DEN(IG)=DEN(IM)                                                PRO4
    DEN(IM)=TEMP                                                   PRO4
    TEMP=RESIST(IG)                                                PRO4
    RESIST(IG)=RESIST(IM)                                          PRO4
    RESIST(IM)=TEMP                                                PRO4
    TEMP=ALPHA(IG)                                                 PRO4
    ALPHA(IG)=ALPHA(IM)                                            PRO4
    ALPHA(IM)=TEMP                                                 PRO4
    TEMP=EMISO(IG)                                                 PRO4
    EMISO(IG)=EMISO(IM)                                            PRO4
    EMISO(IM)=TEMP                                                 PRO4
    TEMP=EMISI(IG)                                                 PRO4
    EMISI(IG)=EMISI(IM)                                            PRO4
    EMISI(IM)=TEMP                                                 PRO4
    TEMP=CONTR(IG)                                                 PRO4
    CONTR(IG)=CONTR(IM)                                            PRO4
    CONTR(IM)=TEMP                                                 PRO4
    GO TO 50                                                       PRO4
 81 IF(IJ-IPR4)82,85,84                                            PRO4
 82 IG=IG+1
    IH=IG+1
    IM=IG
    TMIN=TE(IM)                                                    PRO4
    GO TO 83                                                       PRO4
 84 MACH=81                                                        PRO4
    GO TO 100                                                      PRO4
 85 CONTINUE
    DO 87 I=1,IPR4                                                 PRO4
    IF(IC(I))87,87,88                                              PRO4
 88 IM=IC(I)                                                       PRO4
 89 IF(IZD(IM))90,91,92                                            PRO4
 90 MACH=89                                                        PRO4
    GO TO 100                                                      PRO4
 91 IZD(IM)=I                                                      PRO4
    GO TO 87                                                       PRO4
 92 WRITE(IOUT,93)                                                 PRO4
 93 FORMAT(26HODUPLICATE *4 TABLE NUMBER)
    GO TO 23                                                       PRO4
 87 CONTINUE                                                       PRO4
    IA=1                                                           PRO4
```

```
      DO 95 I=1,30                                                      PRO4
      IH=1                                                              PRO4
      IF(IZC(I))96,97,98                                                PRO4
   96 MACH=95                                                           PRO4
      GO TO 100                                                         PRO4
   97 IF(IZD(I))96,95,99                                                PRO4
   99 IG=IZD(I)                                                         PRO4
      ITABA(I)=IA                                                       PRO4
  104 TEA(IA)=TE(IG)                                                    PRO4
      ALPHAA(IA)=ALPHA(IG)                                              PRO4
      RESIA(IA)=RESIST(IG)                                              PRO4
      THERMA(IA)=THERM(IG)                                              PRO4
      CAPA(IA)=CAP(IG)                                                  PRO4
      DENA(IA)=DEN(IG)                                                  PRO4
      EMISOA(IA)=EMISO(IG)                                              PRO4
      EMISIA(IA)=EMISI(IG)                                              PRO4
      ICA(IA)=IC(IG)                                                    PRO4
      CONTRA(IA)=CONTR(IG)                                              PRO4
      GO TO (102,103,107),IH                                            PRO4
  102 IH=2                                                              PRO4
      IA=IA+1                                                           PRO4
      IG=IG+1                                                           PRO4
      GO TO 104                                                         PRO4
  103 IA=IA+1                                                           PRO4
      IG=IG+1                                                           PRO4
      IF(IC(IG))105,104,107                                             PRO4
  105 IH=3                                                              PRO4
      GO TO 104                                                         PRO4
   98 IF(IZD(I))96,108,99                                               PRO4
  108 IG=IZC(I)                                                         PRO4
      ITABA(I)=IA                                                       PRO4
  114 TEA(IA)=Z(IG+9799)                                                PRO4
      ALPHAA(IA)=Z(IG+9599)                                             PRO4
      THERMA(IA)=Z(IG+9399)                                             PRO4
      RESIA(IA)=Z(IG+9199)                                              PRO4
      DENA(IA)=Z(IG+8799)                                               PRO4
      EMISOA(IA)=Z(IG+8599)                                             PRO4
      EMISIA(IA)=Z(IG+8399)                                             PRO4
      CONTRA(IA)=Z(IG+8199)                                             PRO4
      CAPA(IA)=Z(IG+8999)                                               PRO4
      ICA(IA)=IZA(IG)                                                   PRO4
      GO TO (112,113,107),IH                                            PRO4
  112 IH=2                                                              PRO4
      IA=IA+1                                                           PRO4
      IG=IG+1                                                           PRO4
      GO TO 114                                                         PRO4
  113 IA=IA+1                                                           PRO4
      IG=IG+1                                                           PRO4
      IF(IC(IG))115,114,107                                             PRO4
  115 IH=3                                                              PRO4
      GO TO 114                                                         PRO4
  107 IA=IA+1                                                           PRO4
   95 CONTINUE                                                          PRO4
      IA=IA-1                                                           PRO4
      IF(200-IA)117,118,118                                             PRO4
  117 WRITE(IOUT,119) IA
  119 FORMAT(I4,17HTABLE ENTRIES, *4)
      GO TO 23                                                          PRO4
  118 IPR4=IA                                                           PRO4
      NP4=1                                                             PRO4
      IX=1
```

```
1004 XX=MATC
     XXX=ICA(IX)
     IF(XXX-XX       )1001,1002,1001
1001 IF(IX-200)1003,1005,1005
1003 IX=IX+1
     GO TO 1004
1005 WRITE(IOUT,1006)
1006 FORMAT( 28H0 COLLECTOR MATERIAL MISSING)
1472 LERR=1
     RETURN
1002 IXB(MATC)=IX
1010 IF(ICA(IX))1008,1007,1007
1007 IF(IX-200)1009,1008,1008
1009 IX=IX+1
     GO TO 1010
1008 IXE(MATC)=IX
     IX=1
1014 XX=MATR
     XXX=ICA(IX)
     IF(XXX-XX        )1011,1012,1011
1011 IF(IX-200)1013,1015,1015
1013 IX=IX+1
     GO TO 1014
1015 WRITE(IOUT,1016)
1016 FORMAT( 28H0 RADIATOR MATERIAL MISSING )
     LERR=1
     RETURN
1012 IXB(MATR)=IX
1020 IF(ICA(IX))1018,1017,1017
1017 IF(IX-200)1019,1018,1018
1019 IX=IX+1
     GO TO 1020
1018 IXE(MATR)=IX
     IX=1
1034 XXX=ICA(IX)
     IF(ABS(XXX-ZA(79))-1.E-5)1032,1032,1031
1031 IF(IX-200)1033,1035,1035
1033 IX=IX+1
     GO TO 1034
1035 WRITE(IOUT,1036)
1036 FORMAT( 27H0 HOT SHOE MATERIAL MISSING)
     GO TO 1472
1032 NONIE=ICA(IX)
     IXB(NONIE)=IX
1040 IF(ICA(IX))1038,1037,1037
1037 IF(IX-200)1039,1038,1038
1039 IX=IX+1
     GO TO 1040
1038 IXE(NONIE)=IX
     IX=1
1054 XXX=ICA(IX)
     IF(ABS(XXX-ZA(119))-1.E-5)1052,1052,1051
1051 IF(IX-200)1053,1055,1055
1053 IX=IX+1
     GO TO 1054
1055 WRITE(IOUT,1056)
1056 FORMAT(28H0 COLD SHOE MATERIAL MISSING)
     GO TO 1472
1052 NONIE=ICA(IX)
     IXB(NONIE)=IX
1060 IF(ICA(IX))1058,1057,1057
```

```
1057 IF(IX-200)1059,1058,1058
1059 IX=IX+1
     GO TO 1060
1058 IXE(NONIE)=IX
1303 CONTINUE
     RETURN                                                      PR04
     END
```

```
$IBFTC PRO5    LIST,DECK,DD,XR7
      SUBROUTINE PRO5
C
      COMMON Z
      DIMENSION Z(16000),I5(730),N5(730),P5(730),TIME(100),ISUR(10),
     2T(100),F1C(100),F1R(100),F2C(100),F2R(100),DELTA(100),TSUR(10)
      DIMENSION IXB(20),IXE(20)
      EQUIVALENCE (Z(3656),TIME(1)),(Z(8459),ISUR(1)),(Z(3756),T(1)),
     2(Z(3856),F1C(1)),(Z(3956),F1R(1)),(Z(4056),F2C(1)),(Z(4156),F2R(1)
     3),(Z(4256),DELTA(1)),(Z(6991),N5(1)),(Z(6991),P5(1)),(Z(7721),
     4I5(1)),(Z(2),IOUT),(Z(11),LERR),(Z(8451),I),(Z(8452),N),(Z(8453),
     5NA),(Z(8454),TEMP),(Z(4356),OLDSOL),(Z(4357),ALBEDO),(Z(4358),
     6STEFAN),(Z(8455),TMIN),(Z(8456),J),(Z(4359),ITIME),(Z(8458),K),
     7(Z(831),J5),(Z(4360),TSUR(1))
      EQUIVALENCE (Z(17416),IXB(1)),(Z(17436),IXE(1))
      NAMELIST /NAM1/OLDSOL/NAM2/ALBEDO/NAM3/STEFAN/NAM4/TIME/NAM5/ISUR
     1/NAM6/T/NAM7/F1C/NAM8/F1R/NAM9/F2C/NAM10/F2R/NAM11/DELTA
      I=1
      LERR=0
      IF(I5(I))1,3,4
    1 WRITE(IOUT,2)
    2 FORMAT(19H0ILLEGAL *5 CONTROL)
   22 LERR=1
   15 RETURN
    5 I=I+1
      GO TO (11,12,13,34,34,34,34,34,34,34,21),N
    3 N=N5(I)
      GO TO 5
    4 NA=P5(I)
      TEMP=NA
      IF(ABS (TEMP-P5(I))-1.E-7)6,6,1
    6 N=NA
      GO TO5
   11 IF(I5(I))7,8,9
    7 WRITE(IOUT,10)
   10 FORMAT(19H0YOU GOOFED *5 DATA)
      GO TO 22
    8 OLDSOL=N5(I)
   17 IF(J5-I)26,28,16
    9 OLDSOL=P5(I)
      GO TO 17
   16 I=I+1
      IF(I5(I))17,3,4
   12 IF(I5(I))7,18,19
   18 ALBEDO=N5(I)
      GO TO 17
   19 ALBEDO=P5(I)
      GO TO 17
   13 IF(I5(I))7,20,23
   20 STEFAN=N5(I)
      GO TO 17
   23 STEFAN=P5(I)
      GO TO 17
   21 CALL HEAT(TIME,T,F1C,F1R,F2C,F2R,DELTA)
      TMIN=TIME(1)
      DO 24 J=2,100
      IF(TMIN-TIME(J))24,7,25
   25 IF(TIME(J))7,26,7
```

```
   26 ITIME=J-1
      GO TO 27
   24 CONTINUE
      ITIME=100
   27 DO 29 J=2,10
   29 TSUR(J)=ITIME
      GO TO 17
   34 J=N-3
      ISUR(J)=1
   45 K=ISUR(J)+100*(J-1)
      IF(I5(I))41,42,43
   41 ISUR(J)=ISUR(J)-1
      IF(N-4)17,401,17
  401 IXB(17)=1
      IXE(17)=ISUR(J)
      GO TO 17
   42 TIME(K)=N5(I)
      GO TO 44
   43 TIME(K)=P5(I)
   44 I=I+1
      ISUR(J)=ISUR(J)+1
      GO TO 45
   28 DO 50 I=1,7
      IF(ISUR(I))51,51,50
   51 WRITE(IOUT,52)
   52 FORMAT(27H0PROGRAM NEEDS MORE *5 DATA)
      GO TO 22
   50 CONTINUE
      IF(ISUR(1)-2)54,51,54
   54 DO 58 I=2,7
      IF(ISUR(I)-1)51,58,56
   56 IF(ISUR(I)-ISUR(1))59,58,59
   59 WRITE(IOUT,60)
   60 FORMAT(21H0INCONSISTENT *5 DATA)
      GO TO 22
   58 CONTINUE
      RETURN
      END
```

```
$IBFTC PRO6     LIST,DECK,REF,DD,XR7
      SUBROUTINE PRO6
      COMMON Z
      DIMENSION Z(20000),I6(200),P6(200),N6(200)
      EQUIVALENCE (Z(9051),I6(1)),(Z(8451),N6(1)),(Z(8451),P6(1)),
     1(Z(832),J6),(Z(4400),RE),(Z(4401),SEGM),(Z(4402),TIMAX),
     2(Z(4403),TMIN),(Z(4404),TMX),(Z(4405),TIMF),(Z(4406)
     3,TIMPL),(Z(4407),TIMPS),(Z(4408),XINPUT),(Z(4409),TMAP),
     4(Z(4410),ISS),(Z(4411),EPST),(Z(4412),NL),(Z(4413),NS),
     5(Z(4414),TMACH),(Z(4415),MATC),(Z(4416),MATR),(Z(4417),
     6NRSTRT),(Z(4418),TINIT),(Z(11),LERR),(Z(5200),LB),(Z(2),IOUT),
     7(Z(4419),CFACT)
      I=1
      LERR=0
   20 IF(I-J6)105,105,121
   23 WRITE(IOUT,22)
   22 FORMAT(11H0PRO6 ERROR)
      LERR=1
      RETURN
  105 IF(I6(I))23,130,131
  130 NP=N6(I)
      GO TO 132
  131 NP=P6(I)+1.E-5
  132 I=I+1
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,120,121),NP
    1 IF(I6(I))23,30,31
   30 RE=N6(I)
      GO TO 24
   31 RE=P6(I)
   24 I=I+1
      GO TO 20
    2 IF(I6(I))23,32,33
   32 SEGM=N6(I)
      GO TO 24
   33 SEGM= P6(I)
      GO TO 24
    3 IF(I6(I))23,34,35
   34 TIMAX= N6(I)
      GO TO 24
   35 TIMAX=P6(I)
      GO TO 24
    4 IF(I6(I))23,36,37
   36 TMIN= N6(I)
      GO TO 24
   37 TMIN = P6(I)
      GO TO 24
    5 IF(I6(I))23,38,39
   38 TMX=N6(I)
      GO TO 24
   39 TMX=P6(I)
      GO TO 24
    6 IF(I6(I))23,40,41
   40 TIMF =N6(I)
      GO TO 24
   41 TIMF =P6(I)
      GO TO 24
    7 IF(I6(I))23,42,43
   42 CFACT = N6(I)
```

```
      GO TO 24
   43 CFACT = P6(I)
      GO TO 24
    8 IF(I6(I))23,44,45
   44 TIMPL = N6(I)
      GO TO 24
   45 TIMPL=P6(I)
      GO TO 24
    9 IF(I6(I))23,46,47
   46 TIMPS = N6(I)
      GO TO 24
   47 TIMPS =  P6(I)
      GO TO 24
   10 IF(I6(I))23,48,49
   48 XINPUT =N6(I)
      GO TO 24
   49 XINPUT =P6(I)
      GO TO 24
   11 IF(I6(I))23,50,51
   50 TMAP = N6(I)
      GO TO 24
   51 TMAP = P6(I)
      GO TO 24
   12 IF(I6(I))23,52,53
   52 ISS=N6(I)
      GO TO 91
   53 ISS=P6(I) +.01
   91 IF(ISS-1)92,24,92
   92 NP=NP+4
      I=I+1
      GO TO 20
   13 IF(I6(I))23,54,55
   54 EPST = N6(I)
      GO TO 24
   55 EPST = P6(I)
      GO TO 24
   14 IF(I6(I))23,56,57
   56 NL = N6(I)
      GO TO 24
   57 NL = P6(I) +.01
      GO TO 24
   15 IF(I6(I))23,58,59
   58 NS= N6(I)
      GO TO 24
   59 NS = P6(I) +.01
      GO TO 24
   16 IF(I6(I))23,60,61
   60 TMACH = N6(I)
      GO TO 24
   61 TMACH = P6(I)
      GO TO 24
   17 IF(I6(I))23,62,63
   62 MATC = N6(I)
      GO TO 24
   63 MATC = P6(I) +.01
      GO TO 24
   18 IF(I6(I))23,64,65
   64 MATR = N6(I)
      GO TO 24
   65 MATR = P6(I) +.01
      GO TO 24
```

```
 19 IF(I6(I))23,66,67
 66 NRSTRT =N6(I)
    GO TO 94
 67 NRSTRT = P6(I)   +.01
 94 IF(NRSTRT-1)24,95,24
 95 NP=NP+2
    I=I+1
    GO TO 20
120 IF(I6(I))23,68,69
 68 TINIT =N6(I)
    GO TO 24
 69 TINIT = P6(I)
    GO TO 24
121 RETURN
    END
```

```
$IBFTC HEAT     LIST,DECK,DD,XR7
      SUBROUTINE HEAT(XA,XB,XC,XD,XE,XF,XG)
      COMMON Z
      DIMENSION Z(16000)
      EQUIVALENCE (Z(2),IOUT)
      DIMENSION XA(100),XB(100),XC(100),XD(100),XE(100),XF(100),XG(100)
      XA=XB+XC+XD+XE+XF+XG
      WRITE(IOUT,1)
    1 FORMAT(33H0SUBROUTINE HEAT IS NOT AVAILABLE)
      CALL EXIT
      RETURN
      END
$IBFTC UNPAK    LIST,DECK,DD,XR7
      SUBROUTINE UNPAK
      COMMON Z
      DIMENSION Z(16000),KC(8),I3(1600),K3(220),K4(250),I4(1830),KS(250)
     2,JS(10)
      EQUIVALENCE (Z(5379),I3(1)),(Z(829),J3),(Z(6979),IX),(Z(6980),KA),
     2(Z(6981),KB),(Z(6982),KC(1)),(Z(6990),KD),(Z(10250),K3(1)),(Z(830)
     3,J4),(Z(10000),K4(1)),(Z(10000),I4(1)),(Z(7979),KS(1)),(Z(827),
     4JS(1))
      IF(JS(3))313,313,1
    1 KB=J3/8
      IX=0
      DO 310 KA=1,KB
      KD=K3(KA)
      CALL SPREAD(KC(1),KD)
      DO 311 I=1,8
      IX=IX+1
  311 I3(IX)=KC(I)-1
  310 CONTINUE
      KA=KB+1
  314 IF(IX-J3)312,313,313
  312 IX=IX+1
      I3(IX)=K3(KA)
      KA=KA+1
      GO TO 314
  313 IF(JS(4))413,413,2
    2 KB=J4/8+8
      IF(KB-250) 14, 14,315
  315 KB=250
   14 DO 316 I=1,KB
  316 KS(I)=K4(I)
      KB=J4/8
      IX=0
      DO 410 KA=1,KB
      KD=KS(KA)
      CALL SPREAD(KC(1),KD)
      DO 411 I=1,8
      IX=IX+1
  411 I4(IX)=KC(I)-1
  410 CONTINUE
      KA=KB+1
  414 IF(IX-J4)412,413,413
  412 IX=IX+1
      I4(IX)=KS(KA)
      KA=KA+1
      GO TO 414
```

```
  413 RETURN                                                                U
      END                                                                   U
$IBFTC PACK     LIST,DECK,DD,XR7
      SUBROUTINE PACK
      COMMON Z
      DIMENSION Z(16000),KC(8),K4(250),I4(1830),I3(1600),K3(220),JS(10)
      EQUIVALENCE (Z(5379),I3(1)),                (Z(829),J3),(Z(830),J4),(
     2Z(6979),IX),(Z(6980),KA),(Z(6981),KB),(Z(6982),KC(1)),(Z(6990),KD)
     3,(Z(10000),K4(1)),(Z(10250),K3(1)),(Z(10000),I4(1)),(Z(827),JS(1))
      IF(JS(4))303,303,1
    1 KB=J4/8
      IX=0
      DO 300 KA=1,KB
      DO 301 I=1,8
      IX=IX+1
  301 KC(I)=I4(IX)+1
      CALL SQUASH(KC(1),KD)
  300 K4(KA)=KD
      KA=KB+1
  304 IF(IX-J4)302,303,303
  302 IX=IX+1
      K4(KA)=I4(IX)
      KA=KA+1
      GO TO 304
  303 IF(JS(3))308,308,2
    2 KB=J3/8
      IX=0
      DO 305 KA=1,KB
      DO 306 I=1,8
      IX=IX+1
  306 KC(I)=I3(IX)+1
      CALL SQUASH(KC(1),KD)
  305 K3(KA)=KD
      KA=KB+1
  309 IF(IX-J3)307,308,308
  307 IX=IX+1
      K3(KA)=I3(IX)
      KA=KA+1
      GO TO 309
  308 RETURN
      END
$IBMAP CHECK    M94,999
        ENTRY     CHECK
CHECK   CLA*      3,4
        TMI       CHK2
        TZE       CHK1
        CLA       =1
CHK1    STO*      4,4
        TRA       1,4
CHK2    CLS       =1
        STO*      4,4
        TRA       1,4
        END
$IBMAP SQUASH   M94,999
        ENTRY     SQUASH
        ENTRY     SPREAD
SQUASH  SXA       SQU2,4
        CLA       3,4
        ADD       =8
        STA       SQU1
        PXA       0,0
```

```
        AXT     8,4
        ALS     2
SQU1    ORA     **,4
        TIX     *-2,4,1
SQU2    AXT     **,4
        STO*    4,4
        TRA     1,4
SPREAD  SXA     UNSQ2,4
        CLA     3,4
        ADD     =8
        STA     UNSQ1
        LDQ*    4,4
        LGL     20
        AXT     8,4
        PXA     0,0
        LGL     2
UNSQ1   STO     **,4
        TIX     *-3,4,1
UNSQ2   AXT     **,4
        TRA     1,4
        END
```

```
$IBFTC MESH     LIST,DECK,REF,DD,XR7
      SUBROUTINE MESH
      COMMON Z
      DIMENSION Z(20000),PL(8),L(8),PHI(8),DELTH(8),DELTHE(100),CORA(8),
     2CORR(8),LLA(4),THETAX(100),ANGLE(100),KC(100),RIL(100),ROL(100),
     3RI(100),RO(100),ZA(200)
      EQUIVALENCE (Z(5428),TEMPB),(Z(5500),L(1)),(Z(5470),PHI(1)),(Z(546
     22),DELTH(1)),(Z(11916),DELTHE(1)),(Z(5700),CORA(1)),(Z(5708),CORR(
     31)),(Z(5458),LLA(1)),(Z(5716),THETAX(1)),(Z(5816),ANGLE(1)),(Z(
     412016),KC(1)),(Z(5916),RIL(1)),(Z(6016),ROL(1)),(Z(11816),RI(1)),
     5(Z(11716),RO(1)),(Z(5454),TEMPA),(Z(627),ZA(1)),(ZA(158),PL(1)),
     6(Z(5453),TEMPC),(Z(5451),TEMPD),(Z(5452),TEMPE),(Z(5420),TEMPG),
     7(Z(5422),TEMP),              (Z(5427),LA),(Z(5450),NCOUNT),
     8(ZA(1),X),(ZA(2),Y),(ZA(4),UO),(ZA(5),VO),(ZA(6),U),(ZA(7),V),
     9(Z(2),IOUT),(Z(11),LERR),(Z(5419),LC),(Z(5418),LD),(Z(5417),PLARGE
     9)
      EQUIVALENCE (Z(5416),SUM),(Z(5415),J),(Z(5414),I),(Z(5413),LINE),
     2(Z(5412),THETAE),(Z(5411),M),(Z(5410),K),(Z(5409),TEMG),(Z(5408),
     3TEMGA),(Z(5207),RMIN),(Z(5201),R),(Z(5406),THETM),(Z(5202),THETAV)
     4,(Z(5203),DELR),(Z(5405),KA),(Z(5404),KB),(Z(5403),TEMPH),(Z(5402
     5),TEMPJ),(Z(5401),TEMPK),(Z(5400),N),
     7(Z(5200),LB,L22),(Z(5213),L2),(Z(5214),ITOT,IX),
     8(ZA(166),XIMX),(Z(5208),IMAXP),(ZA(12),XLENGT),(Z(5211),DELX)
     9,(Z(5221),IMAX)
      NAMELIST/NAM1/UO,VO,U,V,X,Y,XIMX,XLENGT,PL/NAM2/PHI,DELTH,
     1DELTHE,PL,L/NAM3/PLARGE,LB,DELTHE,NCOUNT/NAM4/CORA,CORR,LLA/
     2 NAM5/R,RMIN,THETAV,DELR/NAM6/RIL,ROL,THETAX
      LERR=0                                                            MESI
      DO 1 I=1,8
      L(I)=PL(I) +.01
    1 CONTINUE
C     COMPUTE ANGLES                                                    MESI
      TEMPA=(UO+U)*(UO+U)                                               MESI
      TEMPB=(VO+V)*(VO+V)                                               MES
      TEMPC=(2.*X-UO-U)**2                                              MES
      TEMPD=(2.*Y-VO-V)**2                                              MES
      PHI(1)=(4.*X*X-TEMPA-2.*TEMPB-TEMPC)/(-2.*SQRT((TEMPA+TEMPB)*(TEMP
     2C+TEMPB)))
      IF(ABS(PHI(1)-1.E-3))601,602,602
  601 PHI(1)=1.57079
      GO TO 603
  602 PHI(1)=ATAN(SQRT(1.-PHI(1)*PHI(1))/PHI(1))                        MES
  603 IF(PHI(1))200,201,201
  200 PHI(1)=3.14159+PHI(1)
  201 CONTINUE
      PHI(2)=(4.*Y*Y-2.*TEMPC-TEMPB-TEMPD)/(-2.*SQRT((TEMPC+TEMPB)*(TEMP
     2D+TEMPC)))
      IF(ABS(PHI(2)-1.E-3))605,606,606
  605 PHI(2)=1.57079
      GO TO 607
  606 PHI(2)=ATAN(SQRT(1.-PHI(2)*PHI(2))/PHI(2))                        MES
  607 IF(PHI(2))202,203,203
  202 PHI(2)=3.14159+PHI(2)
  203 CONTINUE
      PHI(3)=(4.*X*X-2.*TEMPD-TEMPC-TEMPA)/(-2.*SQRT((TEMPD+TEMPC)*(TEMP
     2A+TEMPD)))
      IF(ABS(PHI(3)-1.E-3))609,610,610
  609 PHI(3)=1.57079
      GO TO 611
  610 PHI(3)=ATAN(SQRT(1.-PHI(3)*PHI(3))/PHI(3))                        MES
  611 IF(PHI(3))204,205,205
```

```
  204 PHI(3)=3.14159+PHI(3)
  205 CONTINUE
      PHI(4)=6.283184-PHI(1)-PHI(2)-PHI(3)
      TEMPE=(U-U0)/(V-V0)
      TEMPG=1./TEMPE
      PHI(5)=2.*ATAN(TEMPE)
      PHI(6)=2.*ATAN(TEMPG)
      PHI(7)=PHI(5)
      PHI(8)=PHI(6)
  COMPUTE DELTA THETA
      DO 2 I=1,8
    2 DELTH(I)=PHI(I)/PL(I)
  COMPARE DELTA THETAS AND RE-SET POINTS AND DELTA THETAS AS NECESSARY
      DO 3 I=1,4
      IF(DELTH(I)-DELTH(I+4))3,3,4
    4 PL(I)=PL(I)*DELTH(I)/DELTH(I+4)
      L(I)=PL(I) +.01
      PL(I)=L(I)
    6 DELTH(I)=PHI(I)/PL(I)
    3 CONTINUE
  CALCULATE WORKING DELTA THETAS
      LB=0
      DO 7 J=1,4
      LA=LB+1
      LB=L(J)+LB
      DO 8 I=LA,LB
    8 DELTHE(I)=DELTH(J)
    7 CONTINUE
      IF(LB-100)9,9,10
   10 WRITE(IOUT,11)
   11 FORMAT(32HONUMBER OF POINTS TABLE EXCEEDED)
      LERR=1
      RETURN
    9 NCOUNT=0
   31 LC=1
      PLARGE=0.
      LD=0
      THETAX(1)=DELTHE(1)
      DELTHE(1)=(DELTHE(2)+DELTHE(LB))/4.    +DELTHE(1)/2.
      SUM=DELTHE(1)
      J=1
   18 LA=LC+1
      LC=L(J)+LD
      DO 13 I=LA,LC
      THETAX(I)=DELTHE(I)
      DELTHE(I)=(DELTHE(I+1)+DELTHE(I-1))/4.    +DELTHE(I)/2.
   13 SUM=SUM+DELTHE(I)
      IF(J-3)21,21,20
   21 TEMP=PHI(J)/SUM
      LA=LD+1
      DO 16 I=LA,LC
      DELTHE(I)=TEMP*DELTHE(I)
      IF(PLARGE-ABS(THETAX(I)-DELTHE(I)))14,16,16
   14 PLARGE=ABS(THETAX(I)-DELTHE(I))
   16 CONTINUE
      IF(J-3)17,19,24
   17 LD=LC
   30 J=J+1
      SUM=0.
      GO TO 18
   19 LD=LC-1
```

```
      GO TO 30                                                           MESH
   20 LC=LC+1                                                            MESH
      TEMP=DELTHE(LC)                                                    MESH
      DELTHE(LC)=(DELTHE(1)+DELTHE(LC-1))/4.        +DELTHE(LC)/2.
      IF(PLARGE-ABS(TEMP-DELTHE(LC)))22,23,23                            MESH
   22 PLARGE=ABS(TEMP-DELTHE(LC))                                        MESH
   23 SUM=SUM+DELTHE(LC)                                                 MESH
      LD=LD+1                                                            MESH
      GO TO 21                                                           MESH
   24 IF(PLARGE-1.E-3)25,25,26
   26 IF(NCOUNT-250)27,27,25                                             MESH
   27 NCOUNT=NCOUNT+1                                                    MESH
      GO TO 31                                                           MESH
   25 CONTINUE                                                           MESH
C     CALCULATE ANGLE OF LOWER LEFT T/E CORNER                           MESH
      TEMP=(UO+U)/2.                                                     MESH
      TEMPA=(VO+V)/2.                                                    MESH
      THETAE=ATAN(TEMP/TEMPA)-PHI(5)/2.                                  MESH
C     CALCULATE CORNER LOCATIONS                                         MESH
      CORA(1)=0.
      CORA(2)=PHI(1)
      CORA(3)=PHI(1)+PHI(2)
      CORA(4)=PHI(3)+CORA(3)
      IF(THETAE)41,42,42                                                 MESH
   41 CORA(5)= THETAE+PHI(5)                                             MESH
      CORA(6)=CORA(5)+PHI(6)                                             MESH
      CORA(7)=CORA(6)+PHI(5)                                             MESH
      CORA(8)=6.283184+THETAE                                            MESH
      GO TO 43                                                           MESH
   42 CORA(5)=THETAE                                                     MESH
      CORA(6)=THETAE+PHI(5)                                              MESH
      CORA(7)=CORA(6)+PHI(6)                                             MESH
      CORA(8)=CORA(7)+PHI(5)                                             MESH
   43 CONTINUE                                                           MESH
C     COMPUTE CORNER RADII                                               MESH
      CORR(1)=SQRT(TEMP*TEMP+TEMPA*TEMPA)                                MESH
      CORR(2)=SQRT((X-TEMP)**2+TEMPA**2)                                 MESH
      CORR(3)=SQRT((X-TEMP)**2+(Y-TEMPA)**2)                             MESH
      CORR(4)=SQRT(TEMP**2+(Y-TEMPA)**2)                                 MESH
      CORR(5)=SQRT(((U-UO)**2+(V-VO)**2)/4.)
      CORR(6)=CORR(5)                                                    MESH
      CORR(7)=CORR(5)                                                    MESH
      CORR(8)=CORR(5)                                                    MESH
C     CALCULATE SIDE NUMBERS                                             MESH
      LLA(1)=L(1)                                                        MESH
      LLA(2)=L(2)+LLA(1)                                                 MESH
      LLA(3)=L(3)+LLA(2)                                                 MESH
      LLA(4)=L(4)+LLA(3)                                                 MESH
C     CALCULATE MINIMUM RADIUS                                           MESH
      TEMG=(U-UO)/2.                                                     MESH
      TEMGA=(V-VO)/2.                                                    MESH
      IF(TEMG-TEMGA) 32,32,33                                            ME
   32 RMIN=TEMG                                                          MES
      GO TO 34                                                           MES
   33 RMIN=TEMGA                                                         MES
   34 R=RMIN                                                             MES
C     COMPUTE AVERAGE                   THETAS                           MES
      SUM=0.                                                             MES
      DO 35 I=1,LC                                                       MES
   36 SUM=SUM+DELTHE(I)                                                  MES
   35 CONTINUE                                                           MES
```

```
      TEMPB=LC                                                          MESH
      THETAV=SUM/TEMPB                                                  MESH
C     CHOOSE DELTA R                                                    MESH
   38 DELR=R*THETAV
C     CALCULATE INCREMENT NUMBER                                        MESH
      I=1                                                               MESH
C     SET UP SUBROUTINE                                                 MESH
      CALL POINT(UO,U,TEMPC,TEMPD,0.)                                   MESH
      CALL POINT(VO,V,TEMPC,TEMPD,-1.)                                  MESH
C     CALCULATE RADII AT BACK SIDE OF INCREMENTS ALONG T/E EDGE (RL) AND
C     ALONG PLATE EDGE  (RL)  ALSO COMPUTE THETA AT THE SAME POINTS
      TEMPC=ATAN(TEMPA/TEMP)                                            MESH
      TEMPD=COS(TEMPC)                                                  MESH
      TEMPE=SIN(TEMPC)                                                  MESH
      THETAX(1)=0.                                                      MESH
      N=1                                                               MESH
      TEMPG=VO                                                          MESH
      TEMPH=U                                                           MESH
      TEMPJ=0.                                                          MESH
      TEMPK=X                                                           MESH
      LA=LLA(1)                                                         MESH
      LB=1                                                              MESH
   49 DO 45 K=LB,LA                                                     MESH
      TEMPI=-(SIN(THETAX(K))*TEMPD+COS(THETAX(K))*TEMPE)                MESH
      RIL(K)=(TEMPG-TEMPA)/TEMPI                                        MESH
      ROL(K)=(TEMPJ-TEMPA)/TEMPI                                        MESH
   45 THETAX(K+1)=THETAX(K)+    DELTHE(K)                               MESH
      N=N+1                                                             MESH
      LB=LA+1                                                           MESH
      LA=LLA(N)                                                         MESH
      DO 46 K=LB,LA                                                     MESH
      TEMPI=-COS(THETAX(K))*TEMPD+SIN(THETAX(K))*TEMPE                  MESH
      RIL(K)=(TEMPH-TEMP)/TEMPI                                         MESH
      ROL(K)=(TEMPK-TEMP)/TEMPI                                         MESH
   46 THETAX(K+1)=THETAX(K)+    DELTHE(K)                               MESH
      IF(N-4)48,47,47                                                   MESH
   48 N=N+1                                                             MESH
      LB=LA+1                                                           MESH
      LA=LLA(N)                                                         MESH
      TEMPG=V                                                           MESH
      TEMPH=UO                                                          MESH
      TEMPJ=Y                                                           MESH
      TEMPK=0.                                                          MESH
      GO TO 49                                                          MESH
   47 M=5
      LB=LA
      DO 70 K=1,LB
      IF(M-9)502,501,501
  501 CORAA=CORA(5)+6.283184
      IF(THETAX(K)-CORAA)71,73,72
  502 IF(THETAX(K)-CORA(M))71,73,72
   71 ANGLE(K)=0.
      GO TO 70
   72 IF(K-1)74,74,575
  575 IF(M-9)75,576,576
  576 ANGLE(K-1)=CORAA
      GO TO 577
   75 ANGLE(K-1)=CORA(M)
  577 M=M+1
      GO TO 70
   73 ANGLE(K)=0.
```

```
      M=M+1
   70 CONTINUE
      IF(M-8)74,77,78 .
   77 ANGLE(LB)=CORA(M)
   78 CONTINUE
      IMAX=XIMX
      IMAXP=IMAX+1
      DELX=XLENGT/XIMX
      L2=2*LB
      ITOT=900/L2
      RETURN                                                        MESH
   74 WRITE(IOUT,76)
   76 FORMAT ( 17H0 BLOW UP IN MESH )
      LERR=1
      RETURN
      END                                                           MESH
```

```
$IBFTC MESHA    LIST,DECK,REF,DD,XR7
      SUBROUTINE MESHA(A,CENX,CENY,PLEBAK,PLEIN,NODE,IX,L2)
      COMMON Z                                                          M
      DIMENSION Z(20000),ANGLE(100),RIL(100),ROL(100),ZA(200),THETAX(100
     1),A(IX,L2),CENX(IX,L2),CENY(IX,L2),PLEBAK(IX,L2),PLEIN(IX,L2),
     2 IB(100),IE(100),NODE(IX,L2),CORR(8),XPNT(100),YPNT(100)
     3,XZ(4),YZ(4),AY(6),AX(6),MA(10),CORA(8)
      EQUIVALENCE (Z(5201),R),(Z(5203),DELR),(Z(5816),ANGLE(1)),
     1(Z(5200),LB),(Z(5916),RIL(1)),(Z(6016),ROL(1)),(Z(2),IOUT),
     2(Z(11),LERR),(Z(627),ZA(1)),(ZA(4),UO),(ZA(5),VO),(ZA(6),U),
     3(ZA(7),V),(Z(5716),THETAX(1)),(Z(5202),THETAV),
     5                 (Z(6116),IB(1)),(Z(6216),IE(1)),
     6         (Z(5708),CORR(1)),(Z(11716),XPNT(1))
      EQUIVALENCE (Z(11816),YPNT(1)),(Z(5204),RP),(Z(5205),KRET),
     1(Z(5700),CORA(1)),(Z(5400),NGOA),(Z(5486),YE),(Z(5492),XE   ),
     2(Z(5487),YF),(Z(5405),ITE),(Z(5406),NGOB),(Z(5493),XF),
     3(Z(5407),IY4),(Z(5408),IY3),(Z(5409),IY2),(Z(5410),IY1),
     4(Z(5411),IX4),(Z(5412),IX3),(Z(5413),IX2),(Z(5414),IX1),
     5(Z(5415),NGO),(Z(5416),AA),(Z(5417),CENYA),(Z(5418),CENXA),
     6(Z(5419),MA3),(Z(5420),MA2),(Z(5421),MA1),(Z(5422),JX),
     7(Z(5423),THE3),(Z(5424),THE2),(Z(5425),THE1),(Z(5426),RO3),
     8(Z(5427),RI3),(Z(5428),RO2),(Z(5429),RI2),(Z(5430),RO1)
      EQUIVALENCE (Z(5431),RI1),(Z(5432),MA(1)),(Z(5442),NAA),
     1(Z(5443),MG),            (Z(5445),MDU),(Z(5446),MCU),
     2(Z(5447),MBU),(Z(5448),MAU),(Z(5449),PV),(Z(5450),PU),
     3(Z(5451),YFP),(Z(5452),XFP),(Z(5453),YEP),(Z(5454),XEP),
     4(Z(5455),YDP),(Z(5456),XDP),(Z(5457),YCP),(Z(5458),XCP),
     5(Z(5459),YBP),(Z(5460),XBP),(Z(5461),YAP),(Z(5462),XAP),
     6(Z(5463),KB),(Z(5464),K),(Z(5485),YD),(Z(5491),XD),(Z(5484),YC  ),
     7(Z(5490),XC),(Z(5483),YB),(Z(5489),XB),(Z(5482),YA),
     8(Z(5488),XA),(Z(5473),J),(Z(5474),YZ(1)),(Z(5478),XZ(1)),
     9(Z(5482),AY(1)),(Z(5488),AX(1)),(Z(5444),NG)
      NAMELIST/NAM7/IB,IE,XPNT,YPNT
      REAL LENGTH
      LB2=2*LB
      DO 2901 J=1,LB2
      IB(J)=0
      IE(J)=0
 2901 CONTINUE
      I=2
      GO TO 705
 1703 DELR=RP*THETAV
      R=RP
  705 RP=R+DELR
      J=1
  701 K=J
      IF(I-15)704,704,1026
  704 A(I,J)=.0
      CENX(I,J)=.0
      CENY(I,J)=.0
      PLEBAK(I,J)=.0
      PLEIN(I,J)=.0
      GO TO 102
  104 CALL POINT(R,THETAX(K),XA,YA,1.)
      CALL POINT(R,THETAX(KB),XB,YB,1.)
      CALL POINT(RP,THETAX(KB),XC,YC,1.)
      CALL POINT(RP,THETAX(K),XD,YD,1.)
      GO TO 109
```

```
  102 IF(K-LB)3,4,4
    4 KB=1
      GO TO 104
    3 KB=K+1
      GO TO 104
  109 CONTINUE
      IF(ANGLE(K))300,300,301
  300 KRET=0
      ANGLE(K)=.0
      GO TO 105
  301 IF(R-CORR(5))101,4083,4083
 4083 ANGLE(K)=.0
      GO TO 300
  101 XAP=XA
      KRET=1
      XBP=XB
      XCP=XC
      XDP=XD
      YAP=YA
      YBP=YB
      YCP=YC
      YDP=YD
 4082 CALL POINT(R,ANGLE(K),XEP,YEP,1.)
      PU=(UO+U)/2.
      PV=(VO+V)/2.
      XZ(1)=XEP
      YZ(1)=YEP
      XZ(2)=PU
      YZ(2)=PV
      XZ(3)=XCP
      YZ(3)=YCP
      XZ(4)=XDP
      YZ(4)=YDP
      CALL INTER(XZ,YZ,XFP,YFP)
 4081 RI1=RIL(K)
      RO1=ROL(K)
      RI2=RIL(KB)
      RO2=ROL(KB)
      RI3=CORR(5)
      DO 4001 JX=1,4
      JXX=JX
      IF(CORA(JX)-ANGLE(K))4001,4002,4001
 4001 CONTINUE
 4002 RO3=CORR(JXX)
      THE1=THETAX(K)
      THE2=THETAX(KB)
      THE3=ANGLE(K)
      CC1=.1 *THETAV
 4092 IF(THE3-THE1-CC1   )4083,4083,4093
 4093 IF(THE2-THE3-CC1   )4083,4083,4091
 4091 XB=XEP
      YB=YEP
      XC=XFP
      YC=YFP
      RIL(KB)=RI3
      ROL(KB)=RO3
      THETAX(KB)=THE3
  105 CONTINUE
C  WHERE IS A, 1 IN PLATE, 2 IN T/E, 3 OUTSIDE PLATE (OR NO PLATE)
      IF(R-RIL(K))6,7,9
    6 MAU=2
```

```
      GO TO 8
    7 MAU=1
      GO TO 8
    9 IF(R-ROL(K))7,7,10
   10 MAU=3
C  WHERE IS B
    8 IF(R-RIL(KB))11,12,13
   11 MBU=2
      GO TO 14
   12 MBU=1
      GO TO 14
   13 IF(R-ROL(KB))12,12,15
   15 MBU=3
C  WHERE IS D
   14 IF(RP-RIL(K))16,17,19
   16 MDU=2
      GO TO 18
   17 MDU=1
      GO TO 18
   19 IF(RP-ROL(K))17,17,20
   20 MDU=3
C  WHERE IS C
   18 IF(RP-RIL(KB))21,22,23
   21 MCU=2
      GO TO 24
   22 MCU=1
      GO TO 24
   23 IF(RP-ROL(KB))22,22,25
   25 MCU=3
   24 MZU=MDU+3*(MCU-1)+9*(MBU-1)+27*(MAU-1)
      GO TO (36,37,38,37,37,37,39,37,40,41,37,42,68,37,69,37,37,43,37,37
     2,37,37,37,37,44,37,45,46,47,37,37,37,37,48,49,50,51,52,53,54,55,
     356,57,58,59,37,37,37,37,37,37,70,60,61,37,37,62,37,37,37,37,37,
     463,37,37,64,37,37,65,37,37,66,37,37,37,37,37,37,37,37,67),MZU
   37 WRITE(IOUT,4069) MZU
 4069 FORMAT(63H0BAD BRANCH IN MESH CALC., SUBROUTINE MESHA, STATEMENT 2
     24, MZU=I3)
      LERR=1
      RETURN
  497 NG=1
      MG=100
  504 IF(NAA.LT.2*MG)GO TO 500
      IF(NAA.LT.3*MG)GO TO 501
      IF(NAA.LT.4*MG)GO TO 502
      IF(NAA.LT.5*MG)GO TO 505
      IF(NAA.LT.6*MG)GO TO 506
      MA(NG)=6
      GO TO 503
  505 MA(NG)=4
      GO TO 503
  506 MA(NG)=5
      GO TO 503
  502 MA(NG)=3
      GO TO 503
  501 MA(NG)=2
      GO TO 503
  500 MA(NG)=1
  503 NAA=NAA-MA(NG)*MG
      MG=MG/10
      NG=NG+1
      IF(NG.LE.3) GO TO 504
```

```
      MA1=MA(1)
      MA2=MA(2)
      MA3=MA(3)
      CALL AREA(AX(MA1),AX(MA2),AX(MA3),AY(MA1),AY(MA2),AY(MA3),AA,CENXA
     2,CENYA)
      A(I,J)=A(I,J)+AA
      CENX(I,J)=CENX(I,J) + CENXA*AA
      CENY(I,J)=CENY(I,J) + CENYA*AA
      GO TO NGO,(510,511,512,513,600,515,517)
C  CASE 8
   36 IX1=1
      IX2=4
      IX3=1
      IX4=2
      NODE(I,J)=5
      ASSIGN 901 TO NGOB
   36 ASSIGN 510 TO NGO
      NAA=123
      GO TO 497
  510 ASSIGN 511 TO NGO
      NAA=134
      GO TO 497
  511 IY1=IX1
      IY2=IX2
      IY3=IX3
      IY4=IX4
      ZZX        =LENGTH(AX(IX1),AY(IY1),AX(IX2),AY(IY2))
      PLEBAK(I,J)=ZZX+PLEBAK(I,J)
      ZZX        = LENGTH(AX(IX3),AY(IY3),AX(IX4),AY(IY4))
      PLEIN(I,J)=ZZX+PLEIN(I,J)
      GO TO 600
C  CASE 1
   51 IX1=1
      IX2=4
      IX3=1
      IX4=2
      NODE(I,J)=1
      ASSIGN 901 TO NGOB
      ITE=1
  950 IF(ITE)1002,1001,1002
 1001 CALL POINT (ROL(K),THETAX(K),XA,YA,1.)
      CALL POINT (ROL(KB),THETAX(KB),XB,YB,1.)
      GO TO 836
 1002 CALL POINT (RIL(K),THETAX(K),XA,YA,1.)
      CALL POINT (RIL(KB),THETAX(KB),XB,YB,1.)
      GO TO 836
C  CASE 2
   46 IX1=6
      IX2=4
      IX3=6
      IX4=5
      NODE(I,J)=1
      ASSIGN 901 TO NGOB
      ITE=1
  846 ASSIGN 512 TO NGO
      NAA=235
  951 IF(ITE)1003,1004,1003
 1004 CALL POINT (ROL(K),THETAX(K),XF,YF,1.)
      CALL POINT (ROL(KB),THETAX(KB),XZ(1),YZ(1),1.)
      GO TO 516
 1003 CALL POINT (RIL(K),THETAX(K),XF,YF,1.)
```

```
      CALL POINT (RIL(KB),THETAX(KB),XZ(1),YZ(1),1.)
  516 XZ(2)=XF
      YZ(2)=YF
      XZ(3)=XA
      YZ(3)=YA
      XZ(4)=XB
      YZ(4)=YB
  618 CALL INTER(XZ,YZ,XE,YE)
      GO TO 497
  512 ASSIGN 513 TO NGO
      NAA=356
      GO TO 497
  513 ASSIGN 511 TO NGO
      NAA=346
      GO TO 497
C   CASE 3
   41 IX1=1
      IX2=4
      IX3=5
      IX4=6
      NODE(I,J)=1
      ASSIGN 901 TO NGOB
      ITE=1
  841 ASSIGN 515 TO NGO
      NAA=156
  952 IF(ITE)1005,1006,1005
 1006 CALL POINT(ROL(KB),THETAX(KB),XF,YF,1.)
      CALL POINT(ROL(K) ,THETAX(K),XZ(1),YZ(1),1.)
      GO TO 516
 1005 CALL POINT(RIL(KB),THETAX(KB),XF,YF,1.)
      CALL POINT(RIL(K),THETAX(K),XZ(1),YZ(1),1.)
      GO TO 516
  515 ASSIGN 517 TO NGO
      NAA=136
      GO TO 497
  517 ASSIGN 511 TO NGO
      NAA=134
      GO TO 497
C   CASE 4
   54 IX1=4
      IX2=6
      IX3=5
      IX4=6
      NODE(I,J)=1
      ASSIGN 901 TO NGOB
      ITE=1
  854 ASSIGN 511 TO NGO
      NAA=456
  953 IF(ITE)1007,1008,1007
 1008 CALL POINT(ROL(K),THETAX(K),XF,YF,1.)
      CALL POINT(ROL(KB),THETAX(KB),XZ(1),YZ(1),1.)
      GO TO 1009
 1007 CALL POINT(RIL(K),THETAX(K),XF,YF,1.)
      CALL POINT(RIL(KB),THETAX(KB),XZ(1),YZ(1),1.)
 1009 XZ(2)=XF
      YZ(2)=YF
      XZ(3)=XC
      YZ(3)=YC
      XZ(4)=XD
      YZ(4)=YD
      GO TO 618
```

```
      CASE 5
   52 IX1=1
      IX2=1
      IX3=5
      IX4=6
      NODE(I,J)=1
      ASSIGN 901 TO NGOB
      ITE=1
  852 ASSIGN 511 TO NGO
      NAA=356
      IF(ITE)1011,1010,1011
 1010 CALL POINT(ROL(KB),THETAX(KB),XE,YE,1.)
      CALL POINT(ROL(K),THETAX(K),XZ(1),YZ(1),1.)
      GO TO 1012
 1011 CALL POINT(RIL(KB),THETAX(KB),XE,YE,1.)
      CALL POINT(RIL(K),THETAX(K),XZ(1),YZ(1),1.)
 1012 XZ(2)=XE
      YZ(2)=YE
      XZ(3)=XC
      YZ(3)=YC
      XZ(4)=XD
      YZ(4)=YD
      CALL INTER (XZ,YZ,XF,YF)
      GO TO 497
      CASE 6
   47 IX1=1
      IX2=4
      IX3=1
      IX4=2
      NODE(I,J)=1
      ASSIGN 901 TO NGOB
      ITE=1
  847 ASSIGN 520 TO NGOA
  955 IF(ITE) 523,1014, 523
 1014 CALL POINT(ROL(K),THETAX(K),XZ(1),YZ(1),1.)
      CALL POINT(ROL(KB),THETAX(KB),XZ(2),YZ(2),1.)
      GO TO 1013
  523 CALL POINT(RIL(K),THETAX(K),XZ(1),YZ(1),1.)
      CALL POINT(RIL(KB),THETAX(KB),XZ(2),YZ(2),1.)
 1013 XZ(3)=XA
      YZ(3)=YA
      XZ(4)=XB
      YZ(4)=YB
      CALL INTER(XZ,YZ,XE,YE)
      XZ(3)=XC
      YZ(3)=YC
      XZ(4)=XD
      YZ(4)=YD
      CALL INTER(XZ,YZ,XF,YF)
      GO TO NGOA,(520,522)
  520 XA=XE
      YA=YE
      XD=XF
      YD=YF
      GO TO 836
      CASE 7
   68 IX1=1
      IX2=4
      IX3=1
      IX4=2
      NODE(I,J)=1
```

```
      ASSIGN 901 TO NGOB
      ITE=1
  868 ASSIGN 522 TO NGOA
      GO TO 955
  522 XB=XE
      YB=YE
      XC=XF
      YC=YF
      GO TO 836
CASE 1P
   40 IX1=1
      IX2=4
      IX3=1
      IX4=2
      NODE(I,J)=3
      ASSIGN 901 TO NGOB
  840 CALL POINT(ROL(K),THETAX(K),XD,YD,1.)
      CALL POINT(ROL(KB),THETAX(KB),XC,YC,1.)
      GO TO 836
CASE 2P
   45 IX1=1
      IX2=6
      IX3=1
      IX4=5
      NODE(I,J)=3
      ASSIGN 901 TO NGOB
      ITE=0
  845 ASSIGN 511 TO NGO
      NAA=156
      GO TO 951
CASE 3P
   63 IX1=1
      IX2=1
      IX3=2
      IX4=5
      NODE(I,J)=3
      ASSIGN 901 TO NGOB
      ITE=0
  863 ASSIGN 511 TO NGO
      NAA=256
      GO TO 952
CASE 4P
   38 IX1=1
      IX2=6
      IX3=1
      IX4=2
      NODE(I,J)=3
      ASSIGN 902 TO NGOB
      ASSIGN 1836 TO NGOC
      ITE=0
      GO TO 854
CASE 5P
   39 IX1=1
      IX2=4
      IX3=1
      IX4=2
      NODE(I,J)=3
      ASSIGN 902 TO NGOB
      ASSIGN 1836 TO NGOC
      ITE=0
      GO TO 852
```

```
      CASE 6P
   44 IX1=1
      IX2=4
      IX3=1
      IX4=2
      NODE(I,J)=3
      ASSIGN 901 TO NGOB
      ITE =0
      GO TO 868
      CASE 7P
   62 IX1=1
      IX2=4
      IX3=1
      IX4=2
      NODE(I,J)=3
      ASSIGN 901 TO NGOB
      ITE=0
      GO TO 847
      CASE  3-4PP
   42 IX1=4
      IX2=6
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ITE=0
      ASSIGN 970 TO NGOC
      GO TO 854
      CASE  3-1PP
   43 IX1=4
      IX2=6
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ASSIGN 970 TO NGOC
      ITE=0
      GO TO 950
      CASE 2-5PP
   48 IX1=1
      IX2=1
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ASSIGN 46  TO NGOC
      ITE =0
      GO TO 852
      CASE 6-5PP
   49 IX1=1
      IX2=1
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ITE=0
      ASSIGN 47 TO NGOC
      GO TO 852
      CASE 2-1PP
   50 IX1=4
      IX2=6
```

```
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ASSIGN 972 TO NGOC
      ITE=0
      GO TO 950
C  CASE 1-4PP
   53 IX1=6
      IX2=4
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ASSIGN 973 TO NGOC
      ITE=0
      GO TO 854
C  CASE  ALL T/E
   55 NODE(I,J)=-1
      ASSIGN 901 TO NGOB
      GO TO 600
C  CASE 4-4PP
   56 IX1=4
      IX2=6
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ITE=0
      ASSIGN 974 TO NGOC
      GO TO 854
C  CASE 1-5PP
   57 IX1=1
      IX2=1
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ITE=0
      ASSIGN 51 TO NGOC
      GO TO 852
C  CASE 5-5PP
   58 IX1=1
      IX2=1
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ASSIGN  52 TO NGOC
      ITE=0
      GO TO 852
C  CASE 1-1PP
   59 IX1=4
      IX2=6
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ASSIGN 973 TO NGOC
      ITE=0
      GO TO 950
```

```
      CASE 6-6PP
   60 IX1=1
      IX2=1
      IX3=5
      IX4=5
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ASSIGN 971 TO NGOC
      ITE=0
      GO TO 847
      CASE 2-2PP
   61 IX1=4
      IX2=6
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ASSIGN 972 TO NGOC
      ITE=0
      GO TO 846
      CASE 3-7PP
   64 IX1=1
      IX2=1
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ASSIGN 41 TO NGOC
      ITE=0
      GO TO 868
      CASE 7-7PP
   65 IX1=1
      IX2=1
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ASSIGN  68 TO NGOC
      ITE=0
      GO TO 868
      CASE 3-3PP
   66 IX1=1
      IX2=1
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ASSIGN  41 TO NGOC
      ITE=0
      GO TO 841
      CASE  ALL OUTSIDE PLATE
   67 NODE(I,J)=0
      ASSIGN 901 TO NGOB
      GO TO 600
      CASE 7-4PP
   69 IX1=4
      IX2=6
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
```

```
      ASSIGN 974 TO NGOC
      ITE=0
      GO TO 854
C     CASE 2-6PP
   70 IX1=1
      IX2=1
      IX3=1
      IX4=1
      NODE(I,J)=4
      ASSIGN 902 TO NGOB
      ASSIGN  46 TO NGOC
      ITE=0
      GO TO 847
  600 GO TO NGOB,(901,902)
  901 IF(KRET)106,108,106
  106 KRET=0
      XA=XEP
      YA=YEP
      XD=XFP
      YD=YFP
      XB=XBP
      YB=YBP
      XC=XCP
      YC=YCP
      ROL(K)=RO3
      RIL(K)=RI3
      ROL(KB)=RO2
      RIL(KB)=RI2
      THETAX(K)=THE3
      THETAX(KB)=THE2
      PBAK=PLEBAK(I,J)
      NIJP=NODE(I,J)
      GO TO 105
  108 IF(ANGLE(K)) 107,107,1015
 1015 PLEBAK(I,J)=PBAK
      XA=XAP
      YA=YAP
      XD=XDP
      YD=YDP
      RIL(K)=RI1
      ROL(K)=RO1
      THETAX(K)=THE1
      IF(NIJP-4)5024,5025,5024
 5024 IF(NODE(I,J)-4)5026,5025,5026
 5025 NODE(I,J)=4
      GO TO 5027
 5026 IF(NIJP+1)5029,5028,5029
 5028 IF(NODE(I,J)+1)5029,5031,5029
 5031 NODE(I,J)=-1
      GO TO 5027
 5029 NODE(I,J)=1
 5027 CONTINUE
      GO TO 107
  902 A(I,J)=-A(I,J)
      CENX(I,J)=-CENX(I,J)
      CENY(I,J)=-CENY(I,J)
      ASSIGN 901 TO NGOB
      GO TO NGOC,(836,970,971,46,972,973,974,51,41,68,1974,1836)
  970 PLEBAK(I,J)=-PLEBAK(I,J)
      GO TO 41
  971 PLEIN(I,J)= -PLEIN(I,J)
```

```
      GO TO 47
  972 PLEBAK(I,J)=-PLEBAK(I,J)
      GO TO 46
  973 PLEBAK(I,J)=-PLEBAK(I,J)
      GO TO 51
  974 PLEBAK(I,J)=-PLEBAK(I,J)
      GO TO 68
 1974 PLEBAK(I,J)=-PLEBAK(I,J)
      GO TO 54
 1836 IX1=1
      IX2=1
      IX3=1
      IX4=1
      GO TO 836
  107 CENX(I,J)=CENX(I,J)/A(I,J)
      CENY(I,J)=CENY(I,J)/A(I,J)
      IF(IB(J))1016,1017,1016
 1017 IF(NODE(I,J)-1)1019,1018,1018
 1018 IB(J)=I
 1016 IF(NODE(I,J))1021,1020,1021
 1020 IF(IE(J))1021,1022,1021
 1022 IE(J)=I-1
      GO TO 1021
 1019 IF(NODE(I,J))1021,1023,1021
 1023 IB(J)=-1
 1021 IF(I-IB(J))1701,2023,1701
 2023 CALL POINT (RIL(K),THETAX(K),XZ(1),YZ(1),1.)
      CALL POINT (RIL(KB),THETAX(KB),XZ(2),YZ(2),1.)
      XZ(3)=CENX(I,J)
      YZ(3)=CENY(I,J)
      XZ(4)=(UO+U)/2.
      YZ(4)=(VO+V)/2.
      CALL INTER(XZ,YZ,XPNT(J),YPNT(J))
 1701 IF(J-LB)1024,1025,1025
 1024 J=J+1
      GO TO 701
 1025 DO 1026 J=1,LB
      IF(IE(J))1027,1027,1026
 1026 CONTINUE
      RETURN
 1027 I=I+1
      IF(I-IX   )1703,1029,1029
 1029 WRITE(IOUT,1030)
 1030 FORMAT(22H0 TOO MANY MESH POINTS)
      LERR=1
      RETURN
      END
```

```
$IBFTC MESHB    LIST,DECK,REF,DD,XR7
      SUBROUTINE MESHB(NODE,A,CENX,CENY,PLEBAK,PLEIN,IX,L2)
      COMMON Z
      DIMENSION Z(20000),IB(100),NODE(IX,L2),A(IX,L2),CENX(IX,L2),
     2CENY(IX,L2),PLEBAK(IX,L2),PLEIN(IX,L2),IE(100)
      EQUIVALENCE (Z(5200),LB),(Z(2),IOUT),(Z(11),LERR),
     2(Z(6116),IB(1)),
     4                    (Z(6216),IE(1)),(Z(5400),I),(Z(5401),JP),
     5(Z(5402),AMIN),(Z(5403),RATIO),(Z(5404),J),(Z(5405),IFLAG)
      IFLAG=0
      AMIN=.45E-4
   12 J=1
  101 K=J
    2 I=IB(J)
  103 IF(NODE(I+1,J))130,30,130
  130 RATIO= A(I+1,J)/A(I,J)
      IF(RATIO-2.)126,126,26
  126 IF(A(I,J)-AMIN)26,26,403
  403 IF(NODE(I,J)-1)404,405,404
  404 IF(NODE(I,J)-4)3,405,3
  405 IF(NODE(I+1,J)-1)406,26,406
  406 IF(NODE(I+1,J)-4)3,26,3
   26 IF(I-IB(J))3,25,3
   25 CENX(I+1,J)=(A(I+1,J)*CENX(I+1,J)+A(I,J)*CENX(I,J))/(A(I+1,J)+A(I,
     1J))
      CENY(I+1,J)=(A(I+1,J)*CENY(I+1,J)+A(I,J)*CENY(I,J))/(A(I+1,J)+A(I,
     1J))
      A(I+1,J)=A(I+1,J)+A(I,J)
      PLEBAK(I+1,J)=PLEBAK(I+1,J)+PLEBAK(I,J)
      IF(NODE(I+1,J)-1)200,201,200
  200 PLEIN(I+1,J)=PLEIN(I    ,J)
      GO TO 202
  201 PLEIN(I+1,J)=PLEIN(I+1,J)+PLEIN(I,J)
  202 NODE(I+1,J)=NODE(I,J)
      NODE(I,J)=-1
      IB(J)=IB(J)+1
      A(I,J)=.0
      CENX(I,J)=.0
      CENY(I,J)=.0
      PLEBAK(I,J)=.0
      PLEIN(I,J)=.0
      GO TO 3
   30 IF(NODE(I-1,J))31,131,131
  131 RATIO= A(I-1,J)/A(I,J)
      IF(RATIO-2.)135,135,36
  135 IF(A(I,J)-AMIN)35,3,3
   31 IF(J-LB)204,203,204
  203 JP=1
      GO TO 205
  204 JP=J+1
  205 IF(NODE(I,JP))3,3,206
  206 IF(A(I,J)-AMIN)45,3,3
   36 IF(I-IE(J))3,35,3
   35 CENX(I-1,J)=(A(I,J)*CENX(I,J)+A(I-1,J)*CENX(I-1,J))/(A(I,J)+A(I-1,
     1J))
      CENY(I-1,J)=(A(I,J)*CENY(I,J)+A(I-1,J)*CENY(I-1,J))/(A(I,J)+A(I-1,
     1J))
      A(I-1,J)=A(I-1,J)+A(I,J)
```

```
      PLEBAK(I-1,J)=PLEBAK(I,J)+PLEBAK(I-1,J)
      NODE(I-1,J)=NODE(I,J)
      NODE(I,J)=0
      IE(J)=IE(J)-1
      A(I,J)=.0
      CENX(I,J)=.0
      CENY(I,J)=.0
      PLEBAK(I,J)=.0
      PLEIN(I,J)=.0
      GO TO 3
   45 CENX(I,J)=(A(I,J)*CENX(I,J)+A(I,JP)*CENX(I,JP))/(A(I,J)+A(I,JP))
      CENY(I,J)=(A(I,J)*CENY(I,J)+A(I,JP)*CENY(I,JP))/(A(I,J)+A(I,JP))
      PLEIN(I,J)=PLEIN(I,J)+PLEIN(I,JP)
      A(I,J)=A(I,J)+A(I,JP)
      LB=LB-1
      DO 305 J=JP,LB
      I=IB(J)
  304 A(I,J)=A(I,J+1)
      CENX(I,J)=CENX(I,J+1)
      CENY(I,J)=CENY(I,J+1)
      PLEBAK(I,J)=PLEBAK(I,J+1)
      PLEIN(I,J)=PLEIN(I,J+1)
      IF(I-IE(J))303,305,305
  303 I=I+1
      GO TO 304
  305 CONTINUE
    3 IF(I-IE(J))306,1,1
  306 I=I+1
      GO TO 103
    1 IF(J-LB)308,307,307
  308 J=J+1
      GO TO 101
  307 IF(IFLAG-1)310,309,310
  310 IFLAG=1
      GO TO 12
  309 RETURN
      END
```

```
$IBFTC MESHC    LIST,DECK,REF,DD,XR7
      SUBROUTINE MESHC(CONDB,NODE,CONDI,PLEBAK,CENX,CENY,A,T,PLEIN,IX,L2
     1)
      COMMON Z
      DIMENSION Z(20000),CONDB(IX,L2),NODE(IX,L2),IB(100),IE(100),
     1CONDI(IX,L2),PLEBAK(IX,L2),XPNT(100),CENX(IX,L2),CENY(IX,L2),
     2A(IX,L2),T(IX,L2),TTE(100),PLEIN(IX,L2),YPNT(100),ZA(200)
      EQUIVALENCE (Z(2),IOUT),(Z(5200),LB),(Z(11),LERR),
     2                            (Z(6216),IE(1)),(Z(6116),IB(1)),
     3                                         (Z(11716),XPNT(1)),
     4(Z(11816),YPNT(1)),
     5(Z(4418),TIN),(Z(5206),TC),(Z(5207),TR),(Z(5208),IMAXP),
     6(Z(6016),TTE(1)),(Z(627),ZA(1)),(Z (626),TP),(Z(5400),JX),
     7(Z(5401),YZ),(Z(5402),XZ),          (Z(5404),X1),
     8                      (Z(5406),JXP),(Z(5407),JPP),(Z(5408),JPX),
     9(Z(5409),JP),(Z(5410),J),(Z(5411),I),(ZA(75),TPR)
      REAL LENGTH
      DO 302 J=1,L2
      DO 301 I=1,IX
      CONDB(I,J)=.0
      CONDI(I,J)=.0
  301 CONTINUE
  302 CONTINUE
      DO 100 J=1,LB
      JX=J+LB
    1 I=IB(J)
      IF(J-1)102,101,102
  101 JP=LB
      JPX=2*LB
      GO TO 2
  102 JP=J-1
      JPX=JP+LB
    2 CONTINUE
   11 IF(J-LB)104,103,104
  103 JPP=1
      JXP=LB+1
      GO TO 105
  104 JPP=J+1
      JXP=JPP+LB
  105 IF(NODE(I,JPP))106,106,107
  106 CONDB(I,JPP)=.0
      CONDB(I,JXP)=CONDB(I,JPP)
  107 IF(NODE(I,JP))108,108,7
  108 CONDB(I,J)=.0
      CONDB(I,JX  )=CONDB(I,J  )
    9 IF(NODE(I+1,J))10,109,10
  109 CONDI(I+1,J)=.0
      CONDI(I+1,JX)=.0
      GO TO 10
    7 X2=CENX(I,JP)
      Y2=CENY(I,JP)
      X1=CENX(I,J)
      Y1=CENY(I,J)
      XLENTH=LENGTH(X1,Y1,X2,Y2)
      CONDB(I,J  )                =(TP*PLEBAK(I,J))/XLENTH
      CONDB(I,JX)=CONDB(I,J)*TPR/TP
      GO TO 9
   10 IF(I-IB(J))25,4,25
```

```
    4 X1=XPNT(J)
      Y1=YPNT(J)
      GO TO 5
   25 X1=CENX(I-1,J)
      Y1=CENY(I-1,J)
    5 X2=CENX(I,J)
      Y2=CENY(I,J)
      XLENTH=LENGTH(X1,Y1,X2,Y2)
      CONDI(I,J)=(TP*PLEIN(I,J))/XLENTH
      CONDI(I,JX)=CONDI(I,J)*TPR/TP
    6 A(I,JX)=A(I,J)
      T(I,J)=TIN
      T(I,JX)=TIN
   16 IF(I-IE(J))112,111,111
  112 I=I+1
      GO TO 2
  111 IB(JX)=IB(J)
      IE(JX)=IE(J)
  100 CONTINUE
      TC=TIN
      TR=TIN
      DO 200 I=2,IMAXP
  200 TTE(I)=TIN
      RETURN
      END
```

```
$IBFTC AREA    LIST,DECK,REF,DD,XR7
      SUBROUTINE AREA (XA,XB,XC,YA,YB,YC,A,CA,CB)
      DIMENSION XZ(4),YZ(4)
      DIMENSION Z(20000),XP(3),C(4)
      EQUIVALENCE(Z(2),IOUT)
      EQUIVALENCE (Z(5300),M),(Z(5301),J),(Z(5302),C(1)),(Z(5306),XP(1))
     2,(Z(5309),K),(Z(5310),I),(Z(5311),CT),(Z(5312),BT),(Z(5313),AT),
     3(Z(5314),F),(Z(5315),E),(Z(5316),G),(Z(5317),D),(Z(5318),TBAR),
     4(Z(5319),SBAR),(Z(5320),T),(Z(5321),S2),(Z(5322),S1),(Z(5323),COSA
     5),(Z(5324),S),(Z(5325),ZA),(Z(5326),Y),(Z(5327),X)
      COMMON Z
      LOGICAL AA,AB,AC
      REAL LENGTH
      IOUT=6
C COMPUTE LINE LENGTHS
      ZA=LENGTH(XA,YA,XB,YB)
      Y=LENGTH(XA,YA,XC,YC)
      X =LENGTH(XB,YB,XC,YC)
      S=(X+Y+ZA)/2.
C COMPUTE AREA
      A=SQRT(S*(S-X)*(S-Y)*(S-ZA))
C COMPUTE CENTROID IF REQUIRED
      GO TO 2
    1 CONTINUE
  101 FORMAT(1H ,6E15.6)
  102 FORMAT(1H ,4I6)
  201 CONTINUE
      RETURN
    2 COSA=(Y*Y+ZA*ZA-X*X)/(2.*Y*ZA)
      COSC=(X*X+Y*Y-ZA*ZA)/(2.*X*Y)
      IF(COSA)103,103,104
  104 IF(COSC)106,106,105
  103 S1=-ZA*COSA
      S2=Y
      T=ZA*SQRT(ABS(1.-COSA*COSA))
      CF=S2/(S1+S2)
      SBAR=((S2-S1)/3.)*CF
      S1=.0
      GO TO 108
  106 S1=Y
      S2=-X*COSC
      T=X*SQRT(ABS(1.-COSC*COSC))
      CF=S1/(S1+S2)
      SBAR=((S2-S1)/3.)*CF
      S2=.0
      GO TO 108
  105 S1=ZA*COSA
      S2=Y-S1
      T=ZA*SQRT(ABS(1.-COSA*COSA))
      CF=1.
      SBAR=((S2-S1)/3.)*CF
  108 TBAR=(T/3.      )*CF
      D=(S1+SBAR)**2+TBAR**2
      G=(S2-SBAR)**2+TBAR**2
      IF(ABS(XA-XC)-1.E-5)110,110,109
  110 IF(YA-YC)111,121,111
  111 C(1)=(YA*YA-YC*YC-D+G)/(2.*(YA-YC))
      C(2)=C(1)
      BT=-2.*XA
      CT=XA*XA+YA*YA-2.*YA*C(1)+C(1)*C(1)-D
      AT=1.
```

```
      CA=BT*BT-4.*AT*CT
      IF(CA)3,112,112
  112 CA=SQRT(ABS(CA))
      C(3)=(CA-BT)/2.
      C(4)=(-BT-CA)/2.
      GO TO 21
  109 E=(D-G-XA*XA+XC*XC-YA*YA+YC*YC)/(2.*(XC-XA))                      AF
      F=(YA-YC)/(XC-XA)                                                 AF
      AT=F*F+1.                                                         AF
      BT=2.*(F*E-XA*F-YA)                                               AF
      CT=XA*XA-2.*XA*E+E*E+YA*YA-D                                      AF
      CA=BT*BT-4.*AT*CT
      IF(CA)3,4,4
    3 WRITE(IOUT,5)
    5 FORMAT(44H0IMAGINARY NUMBER FAILURE IN SUBROUTINE AREA)           AI
   18 GO TO 1
    4 CA=SQRT(ABS(CA))
      CB=(CA-BT)/2./AT                                                  AI
      CT=(-BT-CA)/2./AT                                                 AI
CHECK FOR BEHAVIOR OF VALUE - IT MUST BE IN RANGE OF POINTS             AI
      I=1                                                               AI
      K=1                                                               AI
      XP(1)=YA                                                          A
      XP(2)=YB                                                          A
      XP(3)=YC                                                          A
      C(1)=CB                                                           A
      C(2)=CT                                                           A
   14 CONTINUE
    7 IF(C(I)-XP(1))8,9,10                                              A
    8 IF(C(I)-XP(2))11,9,9                                              A
   10 IF(C(I)-XP(2))9,9,12                                              A
   11 IF(C(I)-XP(3))13,9,9                                              A
   12 IF(C(I)-XP(3))9,9,13                                              A
   13 GO TO (15,16,26,28),K                                             A
   15 J=0                                                               A
      K=2                                                               A
      I=2                                                               A
      GO TO 14                                                          A
   16 IF(J)20,121,20
  121 WRITE(IOUT,17)
   17 FORMAT(41H0CENTROID TEST FAILURE IN SUBROUTINE AREA)              A
      GO TO 18                                                          A
    9 GO TO (19,22,27,31),K                                             A
   19 K=2                                                               A
      I=2                                                               A
      J=1                                                               A
      GO TO 14                                                          A
   22 IF(J)23,24,23                                                     A
   23 J=-1                                                              A
      GO TO 20                                                          A
   24 J=2                                                               A
   20 I=3                                                               A
      XP(1)=XA                                                          A
      XP(2)=XB                                                          A
      XP(3)=XC                                                          A
      K=3                                                               A
      C(3)=E+F*C(1)                                                     A
      C(4)=E+F*C(2)                                                     A
      GO TO 14                                                          A
   26 K=4                                                               A
      M=0                                                               A
```

```
      I=4                                                              A
      GO TO 14                                                         A
   27 K=4                                                              A
      I=4                                                              A
      M=1                                                              A
      GO TO 14                                                         A
   28 IF(M)29,121,29
   31 IF(M)32,33,32                                                    A
   32 M=-1                                                             A
      GO TO 29                                                         A
   33 M=2                                                              A
   29 IF(M)62,121,63
   62 IF(J)21,121,65
   63 IF(J)65,121,64
   64 IF(M-J)121,66,121
   65 IF(J)66,121,35
   66 CA=C(M+2)
      CB=C(M)
      GO TO 201
   35 CA=C(J+2)                                                        A
      CB=C(J)                                                          A
      GO TO 201
   21 XZ(1)=XB
      YZ(1)=YB
      XZ(2)=XC
      YZ(2)=YC
      XZ(3)=XA
      YZ(3)=YA
      XZ(4)=C(3)
      YZ(4)=C(1)
      CALL INTER(XZ,YZ,XPP,YP)
      NN=1
      CY=C(1)
      CX=C(3)
  151 FORMAT(1H ,4E15.6)
  122 AA=XPP.GE.XB
      AB=XPP.LE.XC
      AC=AA.AND.AB
      IF(AC) GO TO 126
      XX=XPP-XB
      XXXX=XPP-XC
  124 IF((XPP.LE.XB).AND.(XPP.GE.XC)) GO TO 126
      IF(ABS(XPP-XB)-1.E-6)205,205,125
  205 IF(ABS(XPP-XC)-1.E-6)126,126,125
  125 GO TO (140,141),NN
  126 IF((YP.GE.YB).AND.(YP.LE.YC)) GO TO 128
  127 IF((YP.LE.YB).AND.(YP.GE.YC)) GO TO 128
      IF(ABS(YP-YB)-1.E-6)206,206,125
  206 IF(ABS(YP-YC)-1.E-6)128,128,125
  128 IF((CX.GE.XA).AND.(CX.LE.XPP)) GO TO 131
  130 IF((CX.LE.XA).AND.(CX.GE.XPP)) GO TO 131
      GO TO 125
  131 IF((CY.GE.YA).AND.(CY.LE.YP)) GO TO 133
  132 IF((CY.LE.YA).AND.(CY.GE.YP)) GO TO 133
      GO TO 125
  133 GO TO (142,143),NN
  140 J=0
  135 NN=2
      XZ(4)=C(4)
      YZ(4)=C(2)
      CALL INTER(XZ,YZ,XPP,YP)
```

```
      CX=C(4)
      CY=C(2)
      GO TO 122
  141 IF(J)121,121,134
  134 CA=C(3)
      CB=C(1)
      GO TO 201
  142 J=1
      GO TO 135
  143 IF(J-1)136,121,136
  136 CA=C(4)
      CB=C(2)
      GO TO 201
      END
```

```
$IBFTC LENGTH   LIST,DECK,DD,XR7
      REAL FUNCTION LENGTH(XA,YA,XB,YB)
      LENGTH=SQRT((XB-XA)**2+(YB-YA)**2)
      RETURN
      END
$IBFTC INTER    LIST,REF,DD,XR7
      SUBROUTINE INTER(X,Y,XA,YA)
      DIMENSION X(4),Y(4),Z(20000)
      COMMON Z
      EQUIVALENCE (Z(2),IOUT)
      IF(X(1)-X(2))10,11,10
   11 IF(X(3)-X(4))10,4,10
   10 IF(Y(1)-Y(2))1,2,1
    2 IF(Y(3)-Y(4))3,4,3
    4 WRITE (IOUT,5)
    5 FORMAT(17H0FAILURE IN INTER)
      CALL EXIT
    3 YA=Y(1)
      XA=(X(4)*Y(3)-X(3)*Y(4)-(X(4)-X(3))*Y(1))/(Y(3)-Y(4))
    6 RETURN
    1 IF(Y(3)-Y(4))7,8,7
    8 YA=Y(3)
      XA=(X(2)*Y(1)-X(1)*Y(2)-(X(2)-X(1))*Y(3))/(Y(1)-Y(2))
      GO TO 6
    7 YA=((X(4)*Y(3)-X(3)*Y(4))/(Y(3)-Y(4))-(X(2)*Y(1)-X(1)*Y(2))/(Y(1)
     2-Y(2)))/((X(4)-X(3))/(Y(3)-Y(4))-(X(2)-X(1))/(Y(1)-Y(2)))
      XA=(X(2)*Y(1)-X(1)*Y(2)-(X(2)-X(1))*YA)/(Y(1)-Y(2))
      GO TO 6
      END
$IBFTC POINT    LIST,DECK,DD,XR7
      SUBROUTINE POINT(R,T,X,Y,B)
      COMMON Z
      DIMENSION Z(20000)
      EQUIVALENCE (Z(5345),U),(Z(5344),V),(Z(5343),TEMP),(Z(5342),TEMPD)
     2,(Z(5341),TEMPE),(Z(5340),TEMPA)
      IF(B)3,1,2
    1 U=(R+T)/2.
    4 RETURN
    3 V=(R+T)/2.
      TEMP=ATAN(V/U)
      TEMPD=COS(TEMP)
      TEMPE=SIN(TEMP)
      GO TO 4
    2 TEMP=COS(T)
      TEMPA=SIN(T)
      X=-R*(TEMP*TEMPD-TEMPA*TEMPE)+U
      Y=-R*(TEMPA*TEMPD+TEMP*TEMPE)+V
      GO TO 4
      END
```

```
      SUBROUTINE PROP(T,V1,V2,V3,V4,V5,V6,MAY)
      COMMON Z
      DIMENSION Z(20000),I(17),IXB(20),IXE(20)
      EQUIVALENCE (Z(17416),IXB(1)),(Z(17436),IXE(1))
      DATA I(1),I(2),I(3),I(4),I(5),I(6),I(7),I(8),I(9),I(10),I(11),I(12
     1),I(13),I(14),I(15),I(16),I(17) / 2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,
     22/
      IF(MAY-6)17,17,50
   50 IF(MAY-12)37,37,51
   51 IF(MAY-16)7,7,52
   52 IF(MAY-17)53,27,53
   53 WRITE(IOUT,54)
   54 FORMAT(13H0 TABLE ERROR)
      LERR=1
      RETURN
    7 IF(IXB(MAY)-IXE(MAY))407,400,400
  400 IXX=IXB(MAY)+1
      V1=Z(IXX+2644)
      V2=Z(IXX+3044)
      V3=Z(IXX+2444)
      V4=Z(IXX+3444)
      V5=Z(IXX+212)
      RETURN
  407 IXX=IXB(MAY)-1+ I(MAY)
 1006 IF(IXX-IXE(MAY))2001,2001,2002
 2002 IXX=IXX-1
      GO TO 1006
 2001 IF(IXX-IXB(MAY)-1)2003,6,6
 2003 IXX=IXX+1
      GO TO 2001
    6 IF(T - Z(IXX+413 ))1003,2004,1
 2004 IXX=IXX+1
    2 A=0.0
      GO TO 8
    1 IXX=IXX+1
      IF(IXX-IXE(MAY))1006,55,55
   55 IXX=IXE(MAY)+1
      GO TO 2
 1003 IF(IXX-IXE(MAY))2011,2011,2012
 2012 IXX=IXX-1
      GO TO 1003
 2011 IF(IXX-IXB(MAY)-1)2013,3,3
 2013 IXX=IXX+1
      GO TO 2011
    3 IF(T - Z(IXX+413 ))4, 2,5
    4 IXX=IXX-1
      IF(IXX-IXB(MAY))57,57,1006
   57 IXX=IXB(MAY)+1
      GO TO 2
    5 A=(T-Z(IXX+412 ))/(Z(IXX+413 )-Z(IXX+412 ))
    8 V1=Z(IXX+2644) + A*(Z(IXX+2645)-Z(IXX+2644))
      V2=Z(IXX+3044) + A*(Z(IXX+3045)-Z(IXX+3044))
      V3=Z(IXX+2444) + A*(Z(IXX+2445)-Z(IXX+2444))
      V4=Z(IXX+3444) + A*(Z(IXX+3445)-Z(IXX+3444))
      V5=Z(IXX+212 ) + A*(Z(IXX+213 )-Z(IXX+212 ))
      I(MAY)=IXX+1-IXB(MAY)
      RETURN
   17 IF(IXB(MAY)-IXE(MAY))417,401,401
  401 IXX=IXB(MAY)+1
      V1=Z(IXX+1031)
      V2=Z(IXX+1231)
```

```
      V3=Z(IXX+1431)
      V4=Z(IXX+1631)
      V5=Z(IXX+2031)
      RETURN
  417 IXX=IXB(MAY)-1 + I (MAY)
 1016 IF(IXX-IXE(MAY))2014,2014,2015
 2015 IXX=IXX-1
      GO TO 1016
 2014 IF(IXX-IXB(MAY)-1)2016,16,16
 2016 IXX=IXX+1
      GO TO 2014
   16 IF(T - Z(IXX+832 ))1013,2017,11
 2017 IXX=IXX+1
   12 A=0.0
      GO TO 18
   11 IXX=IXX+1
      IF(IXX-IXE(MAY))1016,155,155
  155 IXX= IXE(MAY)+1
      GO TO 12
 1013 IF(IXX-IXE(MAY))2018,2018,2019
 2019 IXX=IXX-1
      GO TO 1013
 2018 IF(IXX-IXB(MAY)-1)2020,13,13
 2020 IXX=IXX+1
      GO TO 2018
   13 IF(T-Z(IXX+831 ))14, 12,15
   14 IXX=IXX-1
      IF(IXX-IXB(MAY))157,157,1016
  157 IXX=IXB(MAY) +1
      GO TO 12
   15 A=(T-Z(IXX+831 ))/(Z(IXX+832 )-Z(IXX+831 ))
   18 V1= Z(IXX+1031) +A*(Z(IXX+1032) -Z(IXX+1031))
      V2= Z(IXX+1231) +A*(Z(IXX+1232) -Z(IXX+1231))
      V3= Z(IXX+1431) +A*(Z(IXX+1432) -Z(IXX+1431))
      V4= Z(IXX+1631) +A*(Z(IXX+1632) -Z(IXX+1631))
      V5= Z(IXX+2031) +A*(Z(IXX+2032) -Z(IXX+2031))
      I(MAY)=IXX+1-IXB(MAY)
      RETURN
   37 IF(IXB(MAY)-IXE(MAY))437,402,402
  402 IXX=IXB(MAY)+1
      V3=Z(IXX+12)
      RETURN
  437 IXX=IXB(MAY)-1+I(MAY)
 1036 IF(IXX-IXE(MAY))2023,2023,2022
 2022 IXX=IXX-1
      GO TO 1036
 2023 IF(IXX-IXB(MAY)-1)2024,36,36
 2024 IXX=IXX+1
      GO TO 2023
   36 IF(T-Z(IXX+413))1033,2021,31
 2021 IXX=IXX+1
   32 A=.0
      GO TO 38
   31 IXX=IXX+1
      IF(IXX-IXE(MAY))1036,355,355
  355 IXX=IXE(MAY) +1
      GO TO 32
 1033 IF(IXX-IXE(MAY))2025,2025,2026
 2026 IXX=IXX-1
      GO TO 1033
 2025 IF(IXX-IXB(MAY)-1)2027,33,33
```

```
 2027 IXX=IXX+1
      GO TO 2025
   33 IF(T-Z(IXX+412))34, 32,35
   34 IXX=IXX-1
      IF(IXX-IXB(MAY))357,357,1036
  357 IXX=IXB(MAY)+1
      GO TO 32
   35 A=(T-Z(IXX+412))/(Z(IXX+413)-Z(IXX+412))
   38 V3=Z(IXX+12) +A*(Z(IXX+13)-Z(IXX+12))
      I(MAY)=IXX+1-IXB(MAY)
      RETURN
   27 IF(IXB(MAY)-IXE(MAY))427,403,403
  403 IXX=IXB(MAY)+1
      V1=Z(IXX+3754)
      V2=Z(IXX+3854)
      V3=Z(IXX+3954)
      V4=Z(IXX+4054)
      V5=Z(IXX+4154)
      V6=Z(IXX+4254)
      RETURN
  427 IXX=IXB(MAY)-1 + I(MAY)
 1026 IF(IXX-IXE(MAY))2028,2028,2029
 2029 IXX=IXX-1
      GO TO 1026
 2028 IF(IXX-IXB(MAY)-1)2030,26,26
 2030 IXX=IXX+1
      GO TO 2028
   26 IF(T- Z(IXX+3655))1023,2031,21
 2031 IXX=IXX+1
   22 A=0.0
      GO TO 28
   21 IXX = IXX +1
      IF(IXX-IXE(MAY))1026,255,255
  255 IXX= IXE(MAY)+1
      GO  TO 22
 1023 IF(IXX-IXE(MAY))2032,2032,2033
 2033 IXX=IXX-1
      GO TO 1023
 2032 IF(IXX-IXB(MAY)-1)2034,23,23
 2034 IXX=IXX+1
      GO TO 2032
   23 IF(T- Z(IXX+3654)) 24, 22,25
   24 IXX = IXX-1
      IF(IXX -IXB(MAY))257,257,1026
  257 IXX = IXB(MAY) +1
      GO TO 22
   25 A=(T-Z(IXX+3654))/(Z(IXX+3655)-Z(IXX+3654))
   28 V1=Z(IXX+3754) + A*(Z(IXX+3755)- Z(IXX+3754))
      V2=Z(IXX+3854) + A*(Z(IXX+3855)- Z(IXX+3854))
      V3=Z(IXX+3954) + A*(Z(IXX+3955)- Z(IXX+3954))
      V4=Z(IXX+4054) + A*(Z(IXX+4055)- Z(IXX+4054))
      V5=Z(IXX+4154) + A*(Z(IXX+4155)- Z(IXX+4154))
      V6=Z(IXX+4254) + A*(Z(IXX+4255)- Z(IXX+4254))
      I(MAY)=IXX+1-IXB(MAY)
      RETURN
      END
$ORIGIN         ETA
```

```
$IBFTC CALCS    DECK,XR7
      SUBROUTINE CALCS(XK,CAP,EPSC1,EPSC2,EPSR1,EPSR2,ALPH,T,A,T4,
     1CONDI,CONDB,TERMF,TERMO,IX,L2,     L22)
      COMMON Z
      DIMENSION Z(20000),XK(IX,L2),XKTE( 100 ),CAP(IX,L2),CAPTE( 100 ),
     1ALPHA(100),RESIST(100),EPSC1(IX ,L22),EPSC2(IX ,L22),EPSR1(IX ,L22
     2),EPSR2(IX ,L22),ALPH(IX,L2),EPSI(100),T(IX,L2),TTE(100),A(IX,L2),
     3T4(IX,L2),CONDI(IX,L2),CONDB(IX,L2),TERMF(IX,L2),TERMO(IX,L2),
     4TP(100),IB(100),IE(100),ZA(200),TMPTR(1000),TPRE(1000) ,
     5WRIT(10,12) ,MATTE(15)
      EQUIVALENCE (Z(2),IOUT),(Z(5200),LB),(Z(11),LERR),
     1(Z(11716),XKTE(1)),(Z(5916),EPSI(1)),(Z(6016),TTE(1)),
     2(Z(11816),CAPTE(1)),(Z(5716),ALPHA(1)),(Z(5816),RESIST(1)),
     3(Z(17316),TP(1)) ,(Z(4414),TMACH) ,(Z(3246),WRIT(1)) ,
     4(ZA(54),XMAT )    ,(Z(4415),MATC),(Z(4416),MATR)
      EQUIVALENCE (Z (6116),IB(1)),(Z (6216),IE(1)),
     1(Z(627),ZA(1)),(ZA(4),UO),(ZA(5),VO),(ZA(6),U),(ZA(7),V),
     2(Z (4356),SOLC),(Z (4357),ALBE),(Z (4358),SIGMA),
     3(Z (4400),RE),(Z(5208),IMAXP),(Z ( 626),TPLC),
     4(Z(5467),TPLAN),(Z(5466),F1C),(Z(5465),F1R),(Z(5464),F2C),
     5(Z(5463),F2R),(Z(5468),DELTA),(Z(5206),TC),(Z(5207),TR),
     6(Z(5462),ALPHAC),(Z(5461),EPSC),(Z(5469),TCP),(Z(5460),CAPC),
     7(Z(5459),DELTMP),(Z(5470),TRP),(Z(5458),F3),(Z(5457),F1),
     8(Z(5456),F2),(Z(4411),EPST),(Z(4402),TIMAX),(Z(4412), NL),
     9(Z(4413), NS),(Z(4419),CFACT),(Z(5455),NIT),(Z(5454),NITMAX)
      EQUIVALENCE                        (Z(5452),TAVC4),(Z(5451),EPSAR),
     1(Z(5450),TAVR4),(Z(5449),EPSAC),(Z(5448),RI ),(Z(5447),E),
     2(Z(5446),VOLT),(Z(5445),CURRNT),(Z(5444),POWR),(Z(4410), ISS),
     3(Z(5442),TI4),(Z(5441),TCP4),(Z(5440),CCR),(Z(5439),RCR),
     4(Z(5438),TIME),(Z(5437),TIML),(Z(5436),TIMS),(Z(4406),TIMPL),
     5(Z(4407),TIMPS),(Z(4405),TIMF),(Z(5480),TERM3),
     6(Z(5435),V1),(Z(5434),V2),(Z(5433),V3),(Z(5432),V4),
     7(Z(5431),V5),(Z(5430),V6),(Z(5429),MAY),(Z(5428),J),
     8(Z(5427),TXX),(Z(5426),I),(Z(5425),JL),(Z(5424),JU) ,
     9(ZA(75),TPLR)
      EQUIVALENCE (Z(5423),TERM2),(Z(5422),TERM6),(Z(5421),TERMR),
     1(Z(5420),EA),(Z(5419),ATOT),(Z(5418),AXT),(Z(5417),JJ),
     2(Z(5416),II),(Z(5478),LP),                   (Z(5414),TERM4),
     3(Z(5413),TERM5),(Z(5412),TERM7),(Z(5411),Q),(Z(5410),TERM1),
     4(Z(5409),LBJ),(Z(5408),IC),(Z(5407),JF),(Z(5473),TEPL),
     5(Z(5476),EPSP),(Z(5406),LONG),(Z(5474),NLINE),(Z(14616),TMPTR(1))
     6,(Z(5405),LL),(Z(5404),LU),(Z(5403),IA),
     7(Z(5472),LMAX),(Z(5471),TPR),(Z(5401),JZ),(Z(5400),JY),
     8 (Z(4403),TMIN),(Z(4404),TMX),(Z(4401),SEGM),
     9(Z(5211),DELX),(ZA(166),XIMAX),(Z(18000),TPRE(1))
      DO 7900 IN=1,10
      DO 7900 JN=1,12
 7900 WRIT(IN,JN)=0.
      QO2=0.
      NITMAX=TMX
      TMULT=TIMF
      TDEC=TMULT/100.
      SLB=LB
      LBNEW=SLB/SEGM+.01
      JJZ=1
      DO 7501 I=1,4
      IF(LBNEW-JJZ)7501,7502,7501
 7501 JJZ=2*JJZ
```

```
      LBNEW=1
 7502 WJZ=LBNEW
      NOSEG=LB/LBNEW
      IF(NOSEG-LB)7503,7504,7504
 7503 IL=IB(1)
      IU=IE(1)
      DO 7505 I=IL,IU
      CONDB(I,1)=0.
 7505 CONDB(I,LB+1)=0.
      IL=IB(NOSEG+1)
      J=LB+NOSEG+1
      IU=IE(J)
      DO 7506 I=IL,IU
      CONDB(I,NOSEG+1)=0.
 7506 CONDB(I,J)=0.
 7504 CONTINUE
      TIMF=0.
      JFLAG=0
      NLINE=0
      JCNT=TMACH
      IMAX=XIMAX
      KX=0
      NIT=0
      DELT=1.0
   25 TERM3=SOLC*ALBE
      TIMS = TIMPS                                                      PRNT
      TIML = TIMPL
      DELTM=10.
  607 NIT=NIT+1
      TIME=0.
      IF(NIT-100)7010,7011,7011
 7010 TIMF=TIMF+TDEC
 7011 JTE=XMAT
      JCOL=MATC
      JRAD=MATR
      JCCON=ZA(79)+1.E-4
      JRCON=ZA(119)+1.E-4
      CALL PROP(TC,V1,CAPC,ALPHAC,EPSC,EPS2C,V6,JCOL)
      CALL PROP(TR,V1,CAPR,ALPHAR,EPSR,EPS2R,V6,JRAD)
      DO 600 J=1,NOSEG
       I=IB(J)
  603 TXX = T(I,J)
      CALL PROP ( TXX,V1,V2,V3,V4,V5,V6,JCOL)
      XK(I,J) =V1
      CAP(I,J)=V2
      ALPH(I,J)=V3
      EPSC1(I,J)=V4
      EPSC2(I,J)=V5
      IF(I-IE(J))601,600,600
  601 I =I+1
      GO TO 603
  600 CONTINUE
      JL= LB+1
      JU=NOSEG+LB
      DO 602 J=JL,JU
      I=IB(J)
  605 TXX = T(I,J)
      CALL PROP(TXX,V1,V2,V3,V4,V5,V6,JRAD)
      XK(I,J) =V1
      CAP(I,J)=V2
      ALPH(I,J)=V3
```

```
      EPSR1(I,J)=V4
      EPSR2(I,J)=V5
      IF(I-IE(J))604,602,602
  604 I=I+1
      GO TO 605
  602 CONTINUE
      DO 606 I=2,IMAXP
      TXX=TTE(I)
      CALL PROP(TXX,V1,V2,V3,V4,V5,V6,JTE)
      ALPHA(I)=V1
      XKTE(I) =V2
      RESIST(I)=V3
      CAPTE(I   )=V4
      EPSI(I)     =V5
  606 CONTINUE
      MAY=17
      CALL PROP(TIME,TPLAN,F1C,F1R,F2C,F2R,DELTA,MAY)
      DELTA=DELTA/57.296
      CALL PROP(TC,V1,V2,CCR,V4,V5,V6,JCCON)
      CALL PROP(TR,V1,V2,RCR,V4,V5,V6,JRCON)
      TERM2=SIGMA*TPLAN*TPLAN*TPLAN*TPLAN
      IF(DELTA-1.5708)810,810,811
  810 TERMC=SOLC*COS(DELTA)
      TERMR=.0
      GOTO 812
  811 TERMR=SOLC*COS(DELTA)
      TERMC=.0
  812 CONTINUE
   21 EA=.0
      EPSAR=.0
      TAVR4=.0
      ATOT =.0
      AXT=.0
      JJ=LB+1
   11 II=IB(JJ)
   12 ATOT=ATOT+A(II,JJ)
      T4(II,JJ)=T(II,JJ)*T(II,JJ)*T(II,JJ)*T(II,JJ)
      AXT=AXT + A(II,JJ)*T4(II,JJ)
      EA=EA+A(II,JJ)*EPSR2(II,JJ)
      IF(II-IE(JJ))101,100,100
  101 II=II+1
      GO TO 12
  100 LP=LB+NOSEG
      IF(JJ-LP)103,102,102
  103 JJ=JJ+1
      GO TO 11
  102 TAVR4=AXT/ATOT
      EPSAR=EA/ATOT
      TAVR=SQRT(TAVR4)
      TAVR=SQRT(TAVR)
   20 EA=.0
      TAVC4=.0
      ATOT= .0
      AXT = .0
      EPSAC=.0
      JJ=1
   13 II=IB(JJ)
   14 ATOT=ATOT+A(II,JJ)
      T4(II,JJ)=T(II,JJ)*T(II,JJ)*T(II,JJ)*T(II,JJ)
      AXT=AXT + A(II,JJ)*T4(II,JJ)
      EA=EA+A(II,JJ)*EPSC2(II,JJ)
```

```
      IF(II-IE(JJ))108,109,109
  108 II=II+1
      GO TO 14
  109 IF(JJ-NOSEG)110,111,111
  110 JJ=JJ+1
      GO TO 13
  111 TAVC4=AXT/ATOT
      EPSAC=EA/ATOT
      TAVC=SQRT(TAVC4)
      TAVC=SQRT(TAVC)
      E=(TTE(2)-TR)*    ALPHA(2)
      DO 380 I=3,IMAXP
  380 E=E+(TTE(I)-TTE(I-1))*.5*(ALPHA(I)+ALPHA(I-1))
      E=E+(TC-TTE(IMAXP))*          ALPHA(IMAXP)
      RI=0.
      DO 381 I=2,IMAXP
  381 RI=RI+DELX*RESIST(I)
      RI=1./((U-UO)*(V-VO))*(RI)+RCR+CCR
      CURRNT=E/(RI+RE)
      VOLT=CURRNT*RE
      POWR=VOLT*CURRNT
      GO TO 802
 7042 DELTM=.9*TIMF/DTMAX
      TC=TPRE(1)+DELTM*(TC-TPRE(1))
      TR=TPRE(2) +DELTM*(TR-TPRE(2))
      LI=2
      DO 5019 I=2,IMAXP
      LI=LI+1
 5019 TTE(I)=TPRE(LI)  +DELTM*(TTE(I)-TPRE(LI))
      LL=1
      LU=NOSEG
      DO 5020 J=  LL,LU
      IL=IB(J)
      IU=IE(J)
      DO 5021 I=IL,IU
      LI=LI+1
 5021 T(I,J)=TPRE(LI)  +DELTM*(T(I,J)-TPRE(LI))
 5020 CONTINUE
      LL=LB+1
      LU=LB+NOSEG
      DO 5022 J=LL,LU
      IL=IB(J)
      IU=IE(J)
      DO 5023 I=IL,IU
      LI=LI+1
 5023 T(I,J)=TPRE(LI)  +DELTM*(T(I,J)-TPRE(LI))
 5022 CONTINUE
      GO TO 2074
  366 IF (NIT - NITMAX)607,367,367
  367 CONTINUE
      JFLAG=1
      LONG=1
 2001 QIC=0.
      QRC=0.
      QIR=0.
      QRR=0.
      QOC=0.
      QOR=0.
      QINC=TERMC*ALPHAC*(U-UO)*(V-VO)+TERM3*F2C*ALPHAC*(U-UO)*(V-VO)
     1+TERM2*F1C*(U-UO)*(V-VO)*EPSC
      QRAC=SIGMA*EPSC*TC**4*(U-UO)*(V-VO)
```

```
      QIN=QINC
 7901 QOUT=QRAC
 7902 QINR=TERMR*ALPHAR*(U-UO)*(V-VO)+TERM3*F2R*ALPHAR*(U-UO)
     1*(V-VO)+TERM2*F1R*(U-UO)*(V-VO)*EPSR
      QRAR=SIGMA*EPSR*TR**4*(U-UO)*(V-VO)
      QIN=QIN+QINR
 7904 QOUT=QOUT+QRAR
      DO 7909 J=1,NOSEG
      IL=IB(J)
      IU=IE(J)
      DO 7910 I=IL,IU
      QIC=QIC+TERMC*A(I,J)*ALPH(I,J)+TERM3*A(I,J)*F2C*ALPH(I,J)
     1+TERM2*A(I,J)*F1C*EPSC1(I,J)
      QX=SIGMA*A(I,J)*EPSC1(I,J)*T(I,J)**4
 7912 QRC=QRC+QX
      QOC=QOC+QX
 7910 CONTINUE
 7909 CONTINUE
      QIC=QIC*WJZ
      QRC=QRC*WJZ
      QOC=QOC*WJZ
      QIN=QIN+QIC
      QINC=QINC+QIC
      QRAC=QRAC+QRC
      QOUT=QOUT+QOC
      LL=LB+1
      LU=LB+NOSEG
      DO 7913 J=LL,LU
      IL=IB(J)
      IU=IE(J)
      DO 7914 I=IL,IU
      QIR=QIR+TERMR*A(I,J)*ALPH(I,J)+ALPH(I,J)*TERM3*A(I,J)*F2R
     1 +TERM2*A(I,J)*F1R*EPSR1(I,J)
      QY=SIGMA*A(I,J)*EPSR1(I,J)*T(I,J)**4
 7916 QOR=QOR+QY
      QRR=QRR+QY
 7914 CONTINUE
 7913 CONTINUE
      QOR=QOR*WJZ
      QRR=QRR*WJZ
      QIR=QIR*WJZ
      QIN=QIN+QIR
      QINR=QINR+QIR
      QRAR=QRAR+QRR
      QOUT=QOUT+QOR
      QAVG=.5*(QIN+QOUT+POWR)
      ERROR=      (QIN-(QOUT+POWR))/QAVG
      EFF=100.*(POWR/QINC)
      KX=KX+1
 8300 WRIT(KX,1)=TIME
      WRIT(KX,2)=DELT
      WRIT(KX,3)=NIT
      WRIT(KX,4)=POWR
      WRIT(KX,5)=CURRNT
      WRIT(KX,6)=ERROR
      WRIT(KX,7)=EFF
      WRIT(KX,8)=TC
      WRIT(KX,9)=TR
      WRIT(KX,10)=TAVR
      WRIT(KX,11)=TAVC
      IF(KX-10)8318,8301,8301
```

```
8318 IF(JFLAG)8302,8302,8301
8301 WRITE(IOUT,8303)(WRIT(KX,1),KX=1,10)
8303 FORMAT(1H0,1X,4HTIME,12X,10F11.4)
     WRITE(IOUT,8305)(WRIT(KX,2),KX=1,10)
8305 FORMAT(1H ,1X,14HTIME INCREMENT,2X,10F11.6)
     WRITE(IOUT,8306)(WRIT(KX,3),KX=1,10)
8306 FORMAT(1H ,1X,13HITERATION NO.,3X,10F11.1)
     WRITE(IOUT,8307)(WRIT(KX,4),KX=1,10)
8307 FORMAT(1H ,1X,5HPOWER,11X,10F11.5)
     WRITE(IOUT,8308)(WRIT(KX,5),KX=1,10)
8308 FORMAT(1H ,1X,7HCURRENT,9X,10F11.5)
     WRITE(IOUT,8309)(WRIT(KX,6),KX=1,10)
8309 FORMAT(1H ,1X,5HERROR,11X,10F11.6)
     WRITE(IOUT,8310)(WRIT(KX,7),KX=1,10)
8310 FORMAT(1H ,1X,10HEFFICIENCY,6X,10F11.6)
     WRITE(IOUT,8312)
     WRITE(IOUT,8311)(WRIT(KX,8),KX=1,10)
8312 FORMAT(1H ,1X,12HTEMPERATURES)
8311 FORMAT(1H ,2X,12HHOT JUNCTION,3X,10F11.4)
     WRITE(IOUT,8313)(WRIT(KX,9),KX=1,10)
8313 FORMAT(1H ,2X,13HCOLD JUNCTION,2X,10F11.4)
     WRITE(IOUT,8314)(WRIT(KX,10),KX=1,10)
8314 FORMAT(1H ,2X,14HRADIATOR (AVG),1X,10F11.4)
     WRITE(IOUT,8315)(WRIT(KX,11),KX=1,10)
8315 FORMAT(1H ,2X,14HABSORBER (AVG),1X,10F11.4)
     KX=0
8302 IF(LONG)2019,2019,2008
2008 WRITE(IOUT,2009)
2009 FORMAT(1H1,52X,12HTEMPERATURES//)
     WRITE(IOUT,2011)
2011 FORMAT(3X,4HNODE)
     WRITE(IOUT,2012)
2012 FORMAT(3X,6HNUMBER,6X,1H0,9X,1H1,9X,1H2,9X,1H3,9X,1H4,9X,1H5,9X,1H
    16,9X,1H7,9X,1H8,9X,1H9)
     L=1
     DO 2050 I=2,IMAXP
     TMPTR(L)=TTE(I)
2050 L=L+1
     IF(LBNEW-1)7600,7600,7601
7601 JJZ=LBNEW/2
     IF(JJZ-3)7602,7602,7603
7603 JJZ=3
7602 JJK=1
     LLR=NOSEG
     NONE=0
7608 LLS=NOSEG+1+NONE
     DO 7604 JJY=1,JJZ
     LLS=(LLS-1-NONE)*2+1+NONE
     DO 7605 J=JJK,LLR
     LL=IB(J)
     LU=IE(J)
     DO 7606 I=LL,LU
     LXY=LLS-J +NONE
7606 T(I,LXY  )=T(I,J)
7605 CONTINUE
7604 LLR=(LLR-NONE)*2+NONE
     IF(JJK-1)7607,7607,7600
7607 LLR=LB+NOSEG
     JJK=LB+1
     NONE=LB
     GO TO 7608
```

```
7600 DO 2051 J=1,LB
     IA=IB(J)
     IC=IE(J)
     DO 2052 I=IA,IC
     TMPTR(L)=T(I,J)
2052 L=L+1
2051 CONTINUE
     TMPTR(L)=TC
     L=L+1
     LL=LB+1
     LU=2*LB
     DO 2053 J=LL,LU
     IA=IB(J)
     IC=IE(J)
     DO 2054 I=IA,IC
     TMPTR(L)=T(I,J)
2054 L=L+1
2053 CONTINUE
     TMPTR(L)=TR
     LMAX=L
     WRITE(IOUT,2014) (TMPTR(L),L=1,9)
2014 FORMAT(5X,2H 0,13X,9F10.2)
     L=10
2016 WRITE(IOUT,2015)L,TMPTR(L),TMPTR(L+1),TMPTR(L+2),TMPTR(L+3),TMPTR(
    1L+4),TMPTR(L+5),TMPTR(L+6),TMPTR(L+7),TMPTR(L+8),TMPTR(L+9)
2015 FORMAT(I7,3X,10F10.2)
     L=L+10
     IF(L-LMAX)2016,2016,2019
2019 CONTINUE
     IF(JFLAG)7018,7018,7019
7019 RETURN
7018 GO TO 366
 802 CONTINUE
      TERM4 = 2.* XKTE(IMAXP)*(U-UO)*(V-VO)/DELX
     TERM5 =.0
     TERM7 =.0
     DO 300 J=1,NOSEG
     I=IB(J)
     TERM5 =XK(I,J)*CONDI(I,J)*T(I,J)    +TERM5
 300 TERM7 =XK(I,J)*CONDI(I,J)    +TERM7
     TERM5=TERM5*WJZ
     TERM7=TERM7*WJZ
     Q = CCR*CURRNT**2-ALPHA(IMAXP)*CURRNT*TC+TERMC*ALPHAC*(U-UO)*(V-
    1VO)+TERM3*F2C*ALPHAC*(U-UO)*(V-VO)+TERM2*F1C*(U-UO)*(V-VO)*EPSC
    2-SIGMA*EPSC*TC**4*(U-UO)*(V-VO)    +QO2
     TCP=TC
6016 TPRE(1)=TCP
6015 CONTINUE
     TC=(Q+ TERM4*TTE(IMAXP)+ TERM5)/(TERM4 + TERM7)
     TC = TCP + CFACT *(TC -TCP)
     DTMAX=ABS(TC-TCP)
     TERM4 = 2.*XKTE(2)*(U-UO)*(V-VO)/DELX
     TERM5 = .0
     TERM7 = .0
     Q=RCR*CURRNT**2 +ALPHA(2)    *CURRNT*TR +TERMR*ALPHAR*(U-UO)*(V-VO)
    1 +TERM3*F2R*ALPHAR*(U-UO)*(V-VO)+TERM2*F1R*(U-UO)*(V-VO)*EPSR
    2 -SIGMA*TR**4*(U-UO)*(V-VO) *EPSR   +QO2
     LL =LB +1
     LU=LB+NOSEG
     DO 310 J=LL,LU
     I=IB(J)
```

```
      TERM5=XK(I,J)*CONDI(I,J)*T(I,J)     +TERM5
  310 TERM7=XK(I,J)*CONDI(I,J)    +TERM7
      TERM5=TERM5*WJZ
      TERM7=TERM7*WJZ
      TRP=TR
 6018 TPRE(2)=TRP
 6017 CONTINUE
      TR =(Q +TERM4*TTE(2) +TERM5)/(TERM4 + TERM7)
      TR = TRP +CFACT*(TR-TRP)
      DTMAXP= ABS(TR-TRP)
      IF(DTMAXP-DTMAX)311,311,312
  312 DTMAX = DTMAXP
  311 CONTINUE
      TCP4=.5*(TAVC4+TAVR4)
      LI=2
      QO2=0.
      DO 320 I=2,IMAXP
      XKB=.5*(XKTE(I)+XKTE(I-1))
      XKT=.5*(XKTE(I)+XKTE(I+1))
      LI=LI+1
      TPR=TTE(I)
 6020 TPRE(LI)=TPR
      IF(I-2)321,321,322
  321 TTE(I-1) =TRP
      XKB=2.*XKTE(I)
      ALPHA(I-1)=ALPHA(2)
  322 IF(I-IMAXP)324,323,323
  323 TTE(I+1)=TCP
      XKT=2.*XKTE(I)
      ALPHA(I+1)=ALPHA(IMAXP)
  324 TI4= TTE(I)*TTE(I)*TTE(I)*TTE(I)
      QO1=-SIGMA*EPSI(I)*2.*(U-UO+V-VO)*DELX*(TI4-TCP4)
      Q=+CURRNT**2*RESIST(I)*DELX/((U-UO)*(V-VO))
     2-CURRNT*TTE(I)*(ALPHA(I)-ALPHA(I-1))    +QO1
      TTE(I)=(1./(XKB+XKT))*(Q*DELX/((U-UO)*(V-VO))+
     1TPRE(LI-1)*XKB+TTE(I+1)*XKT)
      TTE(I)= TPR +CFACT*(TTE(I)-TPR)
      DTMAXP = ABS(TTE(I) -TPR)
      IF(DTMAXP-DTMAX)320,320,326
  326 DTMAX = DTMAXP
  320 QO2=QO2-QO1
      QO2=QO2/2.
      LIP=LI
      DO 7001 J=1,NOSEG
      IL=IB(J)
      IU=IE(J)
      DO 7002 I=IL,IU
      LI=LI+1
 7002 TPRE(LI)=T(I,J)
 7001 CONTINUE
      JL=LB+1
      JU=NOSEG+LB
      DO 7003 J=JL,JU
      IL=IB(J)
      IU=IE(J)
      DO 7004 I=IL,IU
      LI=LI+1
 7004 TPRE(LI)=T(I,J)
 7003 CONTINUE
      LI=LIP
      LBJ=NOSEG
```

C R SS

```
      IC=1
 4071 IF(IC-1)331,330,331
  331 J=LB +1
      GO TO 332
  330 J=1
  332 IF(J-LBJ)333,334,333
  334 IF(IC-1)335,336,335
  333 JF=J+1
      GO TO 337
  336 JF=1
      GO TO 337
  335 JF=LB +1
  337 IF(J-1)339,338,339
  339 IF(J-LB -1)340,338,340
  338 JB=LBJ
      GO TO 341
  340 JB=J-1
  341 IL=IB(J)
      IU=IE(J)
      DO 1031 I=IL,IU
      IF(IC-1)342,343,342
  342 F3=1./(1./EPSR2(I,J) +1./EPSAC -1.)
      F1= F1R
      F2= F2R
      TEPL=TPRE(2)
      TERM1 = TERMR
      EPSP=EPSR1(I,J)
      TAV4 = TAVC4
      GO TO 344
  343 F3=1./(1./EPSC2(I,J) +1./EPSAR -1.)
      F1 =F1C
      F2 =F2C
      TEPL=TPRE(1)
      TERM1 =TERMC
      TAV4  =TAVR4
      EPSP=EPSC1(I,J)
  344  Q=TERM1*A(I,J)*ALPH(I,J) +TERM2*A(I,J)*F1*EPSP    +TERM3*A(I,J) *
     1F2*ALPH(I,J) -SIGMA *A(I,J)*EPSP     *T4(I,J)-SIGMA*A(I,J)*F3
     2*(T4(I,J)-TAV4)
      LI=LI+1
      LIP=LI-(IE(J-1)-IB(J)+1 )
      IF(LBJ-LB)6031,6031,6032
 6031 LIPP=LI+LB-1+IE(1)-IB(LB)
      LLL=2
      LLU=LB-1
      DO 6033 JN=LLL,LLU
 6033 LIPP=LIPP+IE(JN)-IB(JN)
      GO TO 6035
 6032 LXY=2*LB
      LZZ=LB+1
      LIPP=LI+LB-1+IE(LZZ)-IB(LXY)
      LLL=LB+2
      LLU=2*LB-1
      DO 6034 JN=LLL,LLU
 6034 LIPP=LIPP+IE(JN)-IB(JN)
 6035 CONTINUE
      IF(I-IB(J))345,346,345
  346 TERMO(I-1,J)= - CONDI(I,J)*TEPL
      GO TO 5071
  345 TERMO(I-1,J)=-CONDI(I,J)*TPRE(LI-1)
 5071 IF(J-1)347,348,347
```

```
  347 IF(J-LB -1)349,348,349
  348 TERMF(I,JB) = -CONDB(I,J)*TPRE(LIPP)
      TP(I) = T(I,J)
      GO TO 5070
  349 TERMF(I,JB)=-CONDB(I,J)*TPRE(LIP)
 5070 TERMO(I,J)= -CONDI(I+1,J)*T(I+1,J)
      IF(J-LBJ)350,351,350
  351 TERMF(I,J) = -CONDB(I,JF)*TP(I)
      GO TO 2070
  350 TERMF(I,J) = -CONDB(I,JF)*T(I,JF)
 2070 TPR = T(I,J)
 6022 TPRE(LI)=TPR
      COND=CONDI(I+1,J)+CONDB(I,JF)
     1+CONDI(I,J)+CONDB(I,J)
      T(I,J)=Q/(XK(I,J)*COND)+(-TERMO(I,J)-TERMF(I,J)
     1-TERMO(I-1,J)-TERMF(I,JB ))/COND
      T(I,J)=TPR + CFACT*(T(I,J)-TPR)
      DTMAXP=ABS(T(I,J)-TPR)
      IF(DTMAXP-DTMAX)1031,1031,353
  353 DTMAX = DTMAXP
 1031 CONTINUE
 1032 IF(J-LBJ)1033,352,352
 1033 J=J+1
      GO TO 332
  352 IF(IC-1)2061,355,2061
  355 IC=0
      LBJ=NOSEG+LB
      GO TO 4071
 2061 IF(DTMAX-EPST)367,367,6011
 6011 IF(NIT-JCNT)7014,7014,7013
 7013 CFACT=0.7
      TIME=.0
      GO TO 7015
 7014 CFACT=CFACT/DTMAX*TIMF*.9
 7015 IF(DTMAX-TIMF)2074,2074,7042
 2074 JZ=MOD(NIT,NL   )
      IF(JZ)2076,2075,2076
 2075 LONG=1
      GO TO 2001
 2076 JY=MOD(NIT,NS )
      IF(JY)366,2078,366
 2078 LONG=0
      GO TO 2001
      END
```

```
$IBFTC CALCT    DECK,XR7
      SUBROUTINE CALCT(XK,CAP,EPSC1,EPSC2,EPSR1,EPSR2,ALPH,T,A,T4,
     1CONDI,CONDB,TERMF,TERMO,IX,L2,     L22)
      COMMON Z
      DIMENSION Z(20000),XK(IX,L2),XKTE( 100 ),CAP(IX,L2),CAPTE( 100 ),
     1ALPHA(100),RESIST(100),EPSC1(IX ,L22),EPSC2(IX ,L22),EPSR1(IX ,L22
     2),EPSR2(IX ,L22),ALPH(IX,L2),EPSI(100),T(IX,L2),TTE(100),A(IX,L2),
     3T4(IX,L2),CONDI(IX,L2),CONDB(IX,L2),TERMF(IX,L2),TERMO(IX,L2),
     4TP(100),IB(100),IE(100),ZA(200),TMPTR(1000),TPRE(1000) ,
     5WRIT(10,12) ,MATTE(15)
      EQUIVALENCE (Z(2),IOUT),(Z(5200),LB),(Z(11),LERR),
     1(Z(11716),XKTE(1)),(Z(5916),EPSI(1)),(Z(6016),TTE(1)),
     2(Z(11816),CAPTE(1)),(Z(5716),ALPHA(1)),(Z(5816),RESIST(1)),
     3(Z(17316),TP(1)) ,(Z(4414),TMACH) ,(Z(3246),WRIT(1)) ,
     4(ZA(54),XMAT )    ,(Z(4415),MATC),(Z(4416),MATR)
      EQUIVALENCE (Z (6116),IB(1)),(Z (6216),IE(1)),
     1(Z(627),ZA(1)),(ZA(4),UO),(ZA(5),VO),(ZA(6),U),(ZA(7),V),
     2(Z (4356),SOLC),(Z (4357),ALBE),(Z (4358),SIGMA),
     3(Z (4400),RE),(Z(5208),IMAXP),(Z ( 626),TPLC),
     4(Z(5467),TPLAN),(Z(5466),F1C),(Z(5465),F1R),(Z(5464),F2C),
     5(Z(5463),F2R),(Z(5468),DELTA),(Z(5206),TC),(Z(5207),TR),
     6(Z(5462),ALPHAC),(Z(5461),EPSC),(Z(5469),TCP),(Z(5460),CAPC),
     7(Z(5459),DELTMP),(Z(5470),TRP),(Z(5458),F3),(Z(5457),F1),
     8(Z(5456),F2),(Z(4411),EPST),(Z(4402),TIMAX),(Z(4412), NL),
     9(Z(4413), NS),(Z(4419),CFACT),(Z(5455),NIT),(Z(5454),NITMAX)
      EQUIVALENCE                 (Z(5452),TAVC4),(Z(5451),EPSAR),
     1(Z(5450),TAVR4),(Z(5449),EPSAC),(Z(5448),RI ),(Z(5447),E),
     2(Z(5446),VOLT),(Z(5445),CURRNT),(Z(5444),POWR),(Z(4410), ISS),
     3(Z(5442),TI4),(Z(5441),TCP4),(Z(5440),CCR),(Z(5439),RCR),
     4(Z(5438),TIME),(Z(5437),TIML),(Z(5436),TIMS),(Z(4406),TIMPL),
     5(Z(4407),TIMPS),(Z(4405),TIMF),(Z(5480),TERM3),
     6(Z(5435),V1),(Z(5434),V2),(Z(5433),V3),(Z(5432),V4),
     7(Z(5431),V5),(Z(5430),V6),(Z(5429),MAY),(Z(5428),J),
     8(Z(5427),TXX),(Z(5426),I),(Z(5425),JL),(Z(5424),JU) ,
     9(ZA(75),TPLR)
      EQUIVALENCE (Z(5423),TERM2),(Z(5422),TERM6),(Z(5421),TERMR),
     1(Z(5420),EA),(Z(5419),ATOT),(Z(5418),AXT),(Z(5417),JJ),
     2(Z(5416),II),(Z(5478),LP),                 (Z(5414),TERM4),
     3(Z(5413),TERM5),(Z(5412),TERM7),(Z(5411),Q),(Z(5410),TERM1),
     4(Z(5409),LBJ),(Z(5408),IC),(Z(5407),JF),(Z(5473),TEPL),
     5(Z(5476),EPSP),(Z(5406),LONG),(Z(5474),NLINE),(Z(14616),TMPTR(1))
     6,(Z(5405),LL),(Z(5404),LU),(Z(5403),IA),
     7(Z(5472),LMAX),(Z(5471),TPR),(Z(5401),JZ),(Z(5400),JY),
     8 (Z(4403),TMIN),(Z(4404),TMX),(Z(4401),SEGM),
     9(Z(5211),DELX),(ZA(166),XIMAX),(Z(18000),TPRE(1))
      DO 310 IN=1,10
      DO 310 JN=1,12
  310 WRIT(IN,JN)=0.
      QO2=0.
      DELTM=10.
      NITMAX=TMX
      TMULT=TIMF
      TIME=0.
      TDEC=TMULT/100.
      SLB=LB
      LBNEW=SLB/SEGM+.01
      JJZ=1
      DO 7501 I=1,4
      IF(LBNEW-JJZ)7501,7502,7501
 7501 JJZ=2*JJZ
      LBNEW=1
```

```
 7502 WJZ=LBNEW
      NOSEG=LB/LBNEW
      IF(NOSEG-LB)7503,7504,7504
 7503 IL=IB(1)
      IU=IE(1)
      DO 7505 I=IL,IU
      CONDB(I,1)=0.
 7505 CONDB(I,LB+1)=0.
      IL=IB(NOSEG+1)
      J=LB+NOSEG+1
      IU=IE(J)
      DO 7506 I=IL,IU
      CONDB(I,NOSEG+1)=0.
 7506 CONDB(I,J)=0.
 7504 CONTINUE
      TIMF=0.
      JFLAG=0
      NLINE=0
      JCNT=TMACH
      IMAX=XIMAX
      KX=0
      NIT=0
      DELT=1.0
   25 TERM3=SOLC*ALBE
      TIMS = TIMPS                                                      PRNT
      TIML = TIMPL
  607 NIT=NIT+1
      IF(NIT-100)7010,7011,7011
 7010 TIMF=TIMF+TDEC
 7011 JTE=XMAT
      JCOL=MATC
      JRAD=MATR
      JCCON=ZA(79)+1.E-4
      JRCON=ZA(119)+1.E-4
      CALL PROP(TC,V1,CAPC,ALPHAC,EPSC,EPS2C,V6,JCOL)
      CALL PROP(TR,V1,CAPR,ALPHAR,EPSR,EPS2R,V6,JRAD)
      DO 600 J=1,NOSEG
       I=IB(J)
  603 TXX = T(I,J)
      CALL PROP ( TXX,V1,V2,V3,V4,V5,V6,JCOL)
      XK(I,J) =V1
      CAP(I,J)=V2
      ALPH(I,J)=V3
      EPSC1(I,J)=V4
      EPSC2(I,J)=V5
      IF(I-IE(J))601,600,600
  601 I =I+1
      GO TO 603
  600 CONTINUE
      JL= LB+1
      JU=NOSEG+LB
      DO 602 J=JL,JU
      I=IB(J)
  605 TXX = T(I,J)
      CALL PROP(TXX,V1,V2,V3,V4,V5,V6,JRAD)
      XK(I,J) =V1
      CAP(I,J)=V2
      ALPH(I,J)=V3
      EPSR1(I,J)=V4
      EPSR2(I,J)=V5
      IF(I-IE(J))604,602,602
```

```
  604 I=I+1
      GO TO 605
  602 CONTINUE
      DO 606 I=2,IMAXP
      TXX=TTE(I)
      CALL PROP(TXX,V1,V2,V3,V4,V5,V6,JTE)
      ALPHA(I)=V1
      XKTE(I) =V2
      RESIST(I)=V3
      CAPTE(I   )=V4
      EPSI(I)    =V5
  606 CONTINUE
      MAY=17
      CALL PROP(TIME,TPLAN,F1C,F1R,F2C,F2R,DELTA,MAY)
      DELTA=DELTA/57.296
      CALL PROP(TC,V1,V2,CCR,V4,V5,V6,JCCON)
      CALL PROP(TR,V1,V2,RCR,V4,V5,V6,JRCON)
      TERM2=SIGMA*TPLAN*TPLAN*TPLAN*TPLAN
      IF(DELTA-1.5708)810,810,811
  810 TERMC=SOLC*COS(DELTA)
      TERMR=.0
      GOTO 812
  811 TERMR=SOLC*COS(DELTA)
      TERMC=.0
  812 CONTINUE
   21 EA=.0
      EPSAR=.0
      TAVR4=.0
      ATOT =.0
      AXT=.0
      JJ=LB+1
   11 II=IB(JJ)
   12 ATOT=ATOT+A(II,JJ)
      T4(II,JJ)=T(II,JJ)*T(II,JJ)*T(II,JJ)*T(II,JJ)
      AXT=AXT + A(II,JJ)*T4(II,JJ)
      EA=EA+A(II,JJ)*EPSR2(II,JJ)
      IF(II-IE(JJ))101,100,100
  101 II=II+1
      GO TO 12
  100 LP=LB+NOSEG
      IF(JJ-LP)103,102,102
  103 JJ=JJ+1
      GO TO 11
  102 TAVR4=AXT/ATOT
      EPSAR=EA/ATOT
      TAVR=SQRT(TAVR4)
      TAVR=SQRT(TAVR)
   20 EA=.0
      TAVC4=.0
      ATOT= .0
      AXT = .0
      EPSAC=.0
      JJ=1
   13 II=IB(JJ)
   14 ATOT=ATOT+A(II,JJ)
      T4(II,JJ)=T(II,JJ)*T(II,JJ)*T(II,JJ)*T(II,JJ)
      AXT=AXT + A(II,JJ)*T4(II,JJ)
      EA=EA+A(II,JJ)*EPSC2(II,JJ)
      IF(II-IE(JJ))108,109,109
  108 II=II+1
      GO TO 14
```

```
  109 IF(JJ-NOSEG)110,111,111
  110 JJ=JJ+1
      GO TO 13
  111 TAVC4=AXT/ATOT
      EPSAC=EA/ATOT
      TAVC=SQRT(TAVC4)
      TAVC=SQRT(TAVC)
      E=(TTE(2)-TR)*     ALPHA(2)
      DO 380 I=3,IMAXP
  380 E=E+(TTE(I)-TTE(I-1))*.5*(ALPHA(I)+ALPHA(I-1))
      E=E+(TC-TTE(IMAXP))*                ALPHA(IMAXP)
      RI=0.
      DO 381 I=3,IMAXP
  381 RI=RI+DELX*.5*(RESIST(I)+RESIST(I-1))
      RI=1./((U-UO)*(V-VO))*(RI)+RCR+CCR
      CURRNT=E/(RI+RE)
      VOLT=CURRNT*RE
      POWR=VOLT*CURRNT
  801 CONTINUE
      DELTS=DELT
      TERM6=DELT/(TPLC*(U-UO)*(V-VO))
      TERM4=(XKTE(IMAXP)*(U-UO)*(V-VO)*(TTE(IMAXP)-TC)    )*2./DELX
      TERM5=.0
      DO 226 J=1,NOSEG
      I=IB(J)
      TERM5= TERM5 +XK(I,J)*CONDI(I,J)*(T(I,J)-TC)
  226 CONTINUE
      TERM5=TERM5*WJZ
      ALPC=ALPHA(IMAXP)
      Q = CCR*CURRNT**2-ABS(ALPC   )*CURRNT*TC+TERMC*ALPHAC*(U-UO)*(V-
     1VO) +TERM3*F2C*ALPHAC*(U-UO)*(V-VO)+TERM2*F1C*(U-UO)*(V-VO)*EPSC
     2-SIGMA*EPSC*TC**4*(U-UO)*(V-VO)   +QO2
      TPRE(1)=TC
      TC= TC +TERM6/CAPC*(Q +TERM4 +TERM5)
      IF(ABS(TC)-1.0E+05)5025,367,367
 5025 CONTINUE
      DTMAX =ABS(TC-TPRE(1))
      TERM6=DELT/(TPLR*(U-UO)*(V-VO))
      TERM4= 2.*XKTE(2)*(U-UO)*(V-VO)*(TTE(2)-TR)    /DELX
      TERM5=.0
      LL=LB+1
      LU=LB+NOSEG
      DO 251 J=LL,LU
      I=IB(J)
  251 TERM5=TERM5 +XK(I,J)*CONDI(I,J)*(T(I,J)-TR)
      TERM5=TERM5*WJZ
      ALPR=ALPHA(2)
      Q = RCR *CURRNT**2 +ABS(ALPR   )*CURRNT*TR+TERMR*ALPHAR*(U-UO)*(V
     1-VO) +TERM3*F2R*ALPHAR*(U-UO)*(V-VO)+TERM2*F1R*(U-UO)*(V-VO)*EPSR
     2-SIGMA*EPSR*TR**4*(U-UO)*(V-VO) +QO2
      TPRE(2)=TR
      TR= TR+ TERM6/CAPR*(Q+TERM4+TERM5)
      DTMAXP=ABS(TR-TPRE(2))
      IF(DTMAXP-DTMAX)5005,5005,5006
 5006 DTMAX=DTMAXP
 5005 CONTINUE
      TCP4=.5*(TAVC4+TAVR4)
      LI=2
      DO 7000 I=2,IMAXP
      LI=LI+1
 7000 TPRE(LI)=TTE(I)
```

```
      LI=2
      QO2=0.
      DO 200 I=2,IMAXP
      QQ=1.
      QR=1.
      QY=2.
      LI=LI+1
      TXB=TPRE(LI-1)
      TXA=TPRE(LI+1)
      IF(I-2)201,201,202
  201 ALPHA(I-1)=ALPHA(2)
      QQ=2.
      QY=3.
  202 IF(I-IMAXP)204,203,203
  203 TXA      =TPRE(1)
      QR=2.
      QY=3.
      ALPHA(I+1)=ALPHA(IMAXP)
  204 TI4=TTE(I)*TTE(I)*TTE(I)*TTE(I)
      QO1=-SIGMA*EPSI(I)*2.*(U-UO+V-VO)*DELX*(TI4-TCP4)
      Q=CURRNT**2*RESIST(I)*DELX/((U-UO)*(V-VO))
     2-CURRNT*TTE(I)*(ALPHA(I)-ALPHA(I-1))  +QO1
      RHS=-XKTE(I)*(U-UO)*(V-VO)/DELX*(-QQ*TXB-QR*TXA
     1+QY*TPRE(LI))+Q
      TTE(I)=TTE(I)+DELT/(CAPTE(I)*(U-UO)*(V-VO)*DELX)*RHS
 5039 DTMAXP=ABS(TTE(I)-TPRE(LI))
      IF(DTMAXP-DTMAX)5007,5007,5008
 5008 DTMAX=DTMAXP
 5007 QO2=QO2-QO1
      QO2=QO2/2.
  200 CONTINUE
      LBJ=NOSEG
   24 IC=1
   10 IF(IC-1)112,113,112
  112 J=LB +1
      GO TO 4
  113 J=1
    4 IF(J-LBJ)115,114,115
  115 JF=J+1
      GO TO 116
  114 IF(IC-1)118,117,118
  118 JF=LB +1
      GO TO 116
  117 JF=1
  116 IF(J-1)120,121,120
  120 IF(J-LB -1)122,121,122
  121 JB=LBJ
      GO TO 2
  122 JB=J-1
    2 IL=IB(J)
      IU=IE(J)
      DO 1002 I=IL,IU
  129 IF(IC-1)131,130,131
  131 F3=1./(1./EPSR2(I,J)+1./EPSAC-1.)
      TEPL=TPRE(2)
      TERM1=TERMR
      F1=F1R
      F2=F2R
      EPSP=EPSR1(I,J)
      THPL =TPLR
      TAV4=TAVC4
```

```
      GO TO 26
  130 F3=1./(1./EPSC2(I,J)+1./EPSAR-1.)
      TEPL=TPRE(1)
      TERM1=TERMC
      TAV4 = TAVR4
      F1=F1C
      F2=F2C
      EPSP=EPSC1(I,J)
      THPL =TPLC
   26 Q=TERM1*A(I,J)*ALPH(I,J)   +  ALPH(I,J)   * TERM3*A(I,J)*F2 -SIGMA
     1*A(I,J)*EPSP     *T4(I,J)-SIGMA*A(I,J)*F3*(T4(I,J)-TAV4)
     2+TERM2*A(I,J)*F1 *EPSP
      LI=LI+1
      TPRE(LI)=T(I,J)
      IF(I-IB(J))141,140,141
  140 TERMO(I-1,J)=-CONDI(I,J)*(T(I,J)-TEPL)
  141 IF(J-1)143,142,143
  143 IF(J-LB -1)821,142,821
  142 TERMF(I,JB) = -CONDB(I,J)*(T(I,J)-T(I,JB))
      TP(I)= T(I,J)
  821 TERMO(I,J)=CONDI(I+1,J)*(T(I,J)-T(I+1,J))
      IF(J-LBJ)145,144,145
  144 TERMF(I,J)=CONDB(I,JF)*(T(I,J)-TP(I))
      GO TO 2071
  145 TERMF(I,J)=CONDB(I,JF)*(T(I,J)-T(I,JF))
 2071 T(I,J)=T(I,J)+ DELT/CAP(I,J)*(Q+XK(I,J)*(-TERMO(I,J)-TERMF(I,J)
     1+TERMO(I-1,J) + TERMF(I,JB)))/(A(I,J)*THPL)
 5041 DTMAXP=ABS(TPRE(LI)-T(I,J))
      IF(DTMAXP-DTMAX)5009,5009,5010
 5010 DTMAX=DTMAXP
 5009 CONTINUE
 1002 CONTINUE
 1003 IF(J-LBJ)1004,147,147
 1004 J=J+1
      GO TO 4
  147 IF(IC-1)27,149,27
  149 IC=0
      LBJ=LB+NOSEG
      GO TO 10
   27 TIME=TIME+DELT
      GOTO 365
 5013 CONTINUE
 5043 IF(NIT-NITMAX)1006,367,367
 1006 CONTINUE
      JZ=MOD(NIT,NL   )
      IF(JZ)5030,362,5030
 5030 JY=MOD(NIT,NS)
      IF(JY)5031,364,5031
 5031 CONTINUE
      IF(TIME-TIML )361,361,362
  362  LONG=1
      NLINE=0
      TIML= TIML + TIMPL
      TIMS= TIMS + TIMPS
      GO TO 2001
  361 IF(TIME-TIMS)5002,5002,364
  364 LONG=0
      TIMS=TIMS + TIMPS
      GO TO 2001
  365 CONTINUE
      IF(DTMAX-TIMF)5014,5014,5012
```

```
5014 DELT=.9*TIMF/DTMAX*DELT
     GO TO 5013
5012 TIME=TIME-DELT
     DELT=DELTM*DELT
     TIME=TIME+DELT
7042 DELTM=.9*TIMF/DTMAX
     TC=TPRE(1)+DELTM*(TC-TPRE(1))
     TR=TPRE(2) +DELTM*(TR-TPRE(2))
     LI=2
     DO 5019 I=2,IMAXP
     LI=LI+1
5019 TTE(I)=TPRE(LI)  +DELTM*(TTE(I)-TPRE(LI))
     LL=1
     LU=NOSEG
     DO 5020 J=  LL,LU
     IL=IB(J)
     IU=IE(J)
     DO 5021 I=IL,IU
     LI=LI+1
5021 T(I,J)=TPRE(LI)  +DELTM*(T(I,J)-TPRE(LI))
5020 CONTINUE
     LL=LB+1
     LU=LB+NOSEG
     DO 5022 J=LL,LU
     IL=IB(J)
     IU=IE(J)
     DO 5023 I=IL,IU
     LI=LI+1
5023 T(I,J)=TPRE(LI)  +DELTM*(T(I,J)-TPRE(LI))
5022 CONTINUE
     GO TO 5013
5002 IF(TIME-TIMAX)366,366,367
 366 IF (NIT - NITMAX)607,367,367
 367 CONTINUE
     JFLAG=1
     LONG=1
2001 QIC=0.
     QRC=0.
     QIR=0.
     QRR=0.
     QOC=0.
     QOR=0.
     QINC=TERMC*ALPHAC*(U-UO)*(V-VO)+TERM3*F2C*ALPHAC*(U-UO)*(V-VO)
    1+TERM2*F1C*(U-UO)*(V-VO)*EPSC
     QRAC=SIGMA*EPSC*TC**4*(U-UO)*(V-VO)
     QIN=QINC
7900 QOUT=QRAC+CAPC*(U-UO)*(V-VO)*TPLC*(TC-TPRE(1))/DELTS
7902 QINR=TERMR*ALPHAR*(U-UO)*(V-VO)+TERM3*F2R*ALPHAR*(U-UO)
    1*(V-VO)+TERM2*F1R*(U-UO)*(V-VO)*EPSR
     QRAR=SIGMA*EPSR*TR**4*(U-UO)*(V-VO)
     QIN=QIN+QINR
7903 QOUT=QOUT+QRAR+CAPR*(U-UO)*(V-VO)*TPLR*(TR-TPRE(2))/DELTS
7905 LI=2
     DO 7906 I=2,IMAXP
     LI=LI+1
7906 CONTINUE
     DO 7909 J=1,NOSEG
     IL=IB(J)
     IU=IE(J)
     DO 7910 I=IL,IU
     LI=LI+1
```

```
      QIC=QIC+TERMC*A(I,J)*ALPH(I,J)+TERM3*A(I,J)*F2C*ALPH(I,J)
     1+TERM2*A(I,J)*F1C*EPSC1(I,J)
      QX=SIGMA*A(I,J)*EPSC1(I,J)*T(I,J)**4
 7911 QOC=QOC+QX+CAP(I,J)*A(I,J)*TPLC*(T(I,J)-TPRE(LI))/DELTS
      QRC=QRC+QX
 7910 CONTINUE
 7909 CONTINUE
      QIC=QIC*WJZ
      QRC=QRC*WJZ
      QOC=QOC*WJZ
      QIN=QIN+QIC
      QINC=QINC+QIC
      QRAC=QRAC+QRC
      QOUT=QOUT+QOC
      LL=LB+1
      LU=LB+NOSEG
      DO 7913 J=LL,LU
      IL=IB(J)
      IU=IE(J)
      DO 7914 I=IL,IU
      LI=LI+1
      QIR=QIR+TERMR*A(I,J)*ALPH(I,J)+ALPH(I,J)*TERM3*A(I,J)*F2R
     1 +TERM2*A(I,J)*F1R*EPSR1(I,J)
      QY=SIGMA*A(I,J)*EPSR1(I,J)*T(I,J)**4
 7915 QOR=QOR+QY+CAP(I,J)*A(I,J)*TPLR*(T(I,J)-TPRE(LI))/DELTS
      QRR=QRR+QY
 7914 CONTINUE
 7913 CONTINUE
      QOR=QOR*WJZ
      QRR=QRR*WJZ
      QIR=QIR*WJZ
      QIN=QIN+QIR
      QINR=QINR+QIR
      QRAR=QRAR+QRR
      QOUT=QOUT+QOR
      QAVG=.5*(QIN+QOUT+POWR)
      ERROR=     (QIN-(QOUT+POWR))/QAVG
      EFF=100.*(POWR/QINC)
      KX=KX+1
 8300 WRIT(KX,1)=TIME
      WRIT(KX,2)=DELT
      WRIT(KX,3)=NIT
      WRIT(KX,4)=POWR
      WRIT(KX,5)=CURRNT
      WRIT(KX,6)=ERROR
      WRIT(KX,7)=EFF
      WRIT(KX,8)=TC
      WRIT(KX,9)=TR
      WRIT(KX,10)=TAVR
      WRIT(KX,11)=TAVC
      IF(KX-10)8318,8301,8301
 8318 IF(JFLAG)8302,8302,8301
 8301 WRITE(IOUT,8303)(WRIT(KX,1),KX=1,10)
 8303 FORMAT(1H0,1X,4HTIME,12X,10F11.4)
      WRITE(IOUT,8305)(WRIT(KX,2),KX=1,10)
 8305 FORMAT(1H ,1X,14HTIME INCREMENT,2X,10F11.6)
      WRITE(IOUT,8306)(WRIT(KX,3),KX=1,10)
 8306 FORMAT(1H ,1X,13HITERATION NO.,3X,10F11.1)
      WRITE(IOUT,8307)(WRIT(KX,4),KX=1,10)
 8307 FORMAT(1H ,1X,5HPOWER,11X,10F11.5)
      WRITE(IOUT,8308)(WRIT(KX,5),KX=1,10)
```

```
 8308 FORMAT(1H ,1X,7HCURRENT,9X,10F11.5)
      WRITE(IOUT,8309)(WRIT(KX,6),KX=1,10)
 8309 FORMAT(1H ,1X,5HERROR,11X,10F11.6)
      WRITE(IOUT,8310)(WRIT(KX,7),KX=1,10)
 8310 FORMAT(1H ,1X,10HEFFICIENCY,6X,10F11.6)
      WRITE(IOUT,8312)
      WRITE(IOUT,8311)(WRIT(KX,8),KX=1,10)
 8312 FORMAT(1H ,1X,12HTEMPERATURES)
 8311 FORMAT(1H ,2X,12HHOT JUNCTION,3X,10F11.4)
      WRITE(IOUT,8313)(WRIT(KX,9),KX=1,10)
 8313 FORMAT(1H ,2X,13HCOLD JUNCTION,2X,10F11.4)
      WRITE(IOUT,8314)(WRIT(KX,10),KX=1,10)
 8314 FORMAT(1H ,2X,14HRADIATOR (AVG),1X,10F11.4)
      WRITE(IOUT,8315)(WRIT(KX,11),KX=1,10)
 8315 FORMAT(1H ,2X,14HABSORBER (AVG),1X,10F11.4)
      KX=0
 8302 IF(LONG)2019,2019,2008
 2008 WRITE(IOUT,2009)
 2009 FORMAT(1H1,52X,12HTEMPERATURES//)
      WRITE(IOUT,2011)
 2011 FORMAT(3X,4HNODE)
      WRITE(IOUT,2012)
 2012 FORMAT(3X,6HNUMBER,6X,1H0,9X,1H1,9X,1H2,9X,1H3,9X,1H4,9X,1H5,9X,1H
     16,9X,1H7,9X,1H8,9X,1H9)
      L=1
      DO 2050 I=2,IMAXP
      TMPTR(L)=TTE(I)
 2050 L=L+1
      IF(LBNEW-1)7600,7600,7601
 7601 JJZ=LBNEW/2
      IF(JJZ-3)7602,7602,7603
 7603 JJZ=3
 7602 JJK=1
      LLR=NOSEG
      NONE=0
 7608 LLS=NOSEG+1+NONE
      DO 7604 JJY=1,JJZ
      LLS=(LLS-1-NONE)*2+1+NONE
      DO 7605 J=JJK,LLR
      LL=IB(J)
      LU=IE(J)
      DO 7606 I=LL,LU
      LXY=LLS-J +NONE
 7606 T(I,LXY  )=T(I,J)
 7605 CONTINUE
 7604 LLR=(LLR-NONE)*2+NONE
      IF(JJK-1)7607,7607,7600
 7607 LLR=LB+NOSEG
      JJK=LB+1
      NONE=LB
      GO TO 7608
 7600 DO 2051 J=1,LB
      IA=IB(J)
      IC=IE(J)
      DO 2052 I=IA,IC
      TMPTR(L)=T(I,J)
 2052 L=L+1
 2051 CONTINUE
      TMPTR(L)=TC
      L=L+1
      LL=LB+1
```

```
      LU=2*LB
      DO 2053 J=LL,LU
      IA=IB(J)
      IC=IE(J)
      DO 2054 I=IA,IC
      TMPTR(L)=T(I,J)
 2054 L=L+1
 2053 CONTINUE
      TMPTR(L)=TR
      LMAX=L
      WRITE(IOUT,2014) (TMPTR(L),L=1,9)
 2014 FORMAT(5X,2H 0,13X,9F10.2)
      L=10
 2016 WRITE(IOUT,2015)L,TMPTR(L),TMPTR(L+1),TMPTR(L+2),TMPTR(L+3),TMPTR(
     1L+4),TMPTR(L+5),TMPTR(L+6),TMPTR(L+7),TMPTR(L+8),TMPTR(L+9)
 2015 FORMAT(I7,3X,10F10.2)
      L=L+10
      IF(L-LMAX)2016,2016,2019
 2019 CONTINUE
      IF(JFLAG)7018,7018,7019
 7019 RETURN
 7018 GO TO 5002
      END
```